A STUDY OF APPLICATION OF REMOTE SENSING TO RIVER FORECASTING

IBM No. 75W-00056

FINAL REPORT

VOLUME II – DETAILED TECHNICAL REPORT: NASA–IBM STREAMFLOW FORECAST MODEL USER'S GUIDE

(NASA-CR-143859) A STUDY OF APPLICATION OF REMOTE SENSING TO RIVER FORECASTING. VOLUME 2: DETAILED TECHNICAL REPORT, NASA-IBM STREAMFLOW FORECAST MODEL USER'S GUIDE Final Report (International Business

N75-25265

Unclas

G3/43 24705

VOLUME II - DETAILED TECHNICAL REPORT:
NASA-IBM STREAMFLOW FORECAST MODEL USER'S GUIDE

TABLE OF CONTENTS

TABLE OF CONTENTS
(CONTINUED)

SECTION 1

INTRODUCTION

This volume is intended to aid a prospective user to operate the NASA-IBM Streamflow Forecast Model. This includes data preparation, determining model parameters, initializing and optimizing parameters (calibration) selecting control options and interpreting results. Some background information has been included in Section 2. Appendices contain a dictionary of variables, a source program listing and flow charts. A bibliography, with abstracts, several related documents, and computer programs are to be delivered separately.

The model has been operated for this study on an IBM System/360 Model 44, using a Model 2250 keyboard/graphics terminal for interactive operation. The model can be set up and operated in a batch processing mode on any System/360 or 370 that has the memory capacity. The model requires 210K bytes of core storage, and the optimization program, OPSET (which was used previous to but not in this study), requires 240K bytes. The data bank for one small watershed requires approximately 32 tracks of disk storage (Model 2314).

The models described in this report have evolved from the well-known Stanford Watershed Models (Crawford and Linsley, 1962; Crawford and Linsley, 1966; Anderson and Crawford, 1964). These original models were written in the BALGOL computer language for the Burroughs Computer at the Stanford Research Institute. Since BALGOL is not used extensively, a number of FORTRAN translations have been published (James, 1968; Claborn and Moore, 1970; Ricci, 1972, among others). The translation by James, known as the Kentucky Watershed Model, was selected for the previous study and used with some modifications.

The Kentucky Watershed Model is basically a FORTRAN version of the Stanford Watershed Model III (Anderson and Crawford, 1964). A few modifications have been added from the Stanford Watershed Model IV (Crawford and Linsley, 1966) by James for use in Eastern watersheds. A self-calibrating version of the KWM (OPSET) was developed and tested by Liou (1970) and applied by James (1970) and Ross (1970).

Modifications introduced by IBM under NASA contract consist principally of adding routines for man-machine interfaces, multi-year simulations, statistical analysis of results, and plot output options. Additionally, shortened, variable simulation periods have been introduced for forecast applications.

For more detailed discussions of the model structure and application, the reader is referred to reports by Anderson and Crawford (1964), Crawford and Linsley (1966), Liou (1970), Ross (1970), James (1970), Ricci (1972 a, b, c), and Striffler (1973).

SECTION 2

SUMMARY DESCRIPTION

2.1 HYDROLOGIC PHENOMENA IN THE WATERSHED

That aspect of hydrology known as streamflow forecasting undertakes to predict the outflow from a river basin, in terms of flow rate as a function of time, in response to a given precipitation event under given initial conditions. This capability is vital to effective planning for urban/industrial development, flood control, hydroelectric power, navigation, and water resources management.

Figure 2-1 depicts the cross section of a somewhat idealized rural catchment and identifies the principal phenomena at work in the rainfall-runoff relationship. The input (precipitation) is partially intercepted by vegetation and water retention areas. Moisture reaching pervious surfaces divides among overland flow, infiltration, and evaporation. Through subsurface processes, interflow and groundwater flow contribute ultimately to streamflow, with some losses due to transpiration through plant life. In certain regions, in winter, moisture is stored in the form of snow in portions of the basin, and melts to produce additional moisture movement in spring.

All the phenomena involved in this portion of the hydrologic cycle are widely and well understood qualitatively, and several empirical relationships have been developed from a combination of theory and experiment. The relationships are numerous, many of then are nonlinear, and they are interrelated. Manual solutions for streamflow by manipulation of such a set of equations are inefficient and so time consuming as to be of little value in an operational situation. Individuals and organizations responsible for streamflow forecasting have turned to watershed models as effective tools for their work. Development of such models has been facilitated by the increasing availability of large, high-speed computers.

2.2 MOISTURE ACCOUNTING IN THE SIMULATION MODEL

The streamflow forecast model uses a moisture accounting system to synthesize a continuous hydrograph (a graph of streamflow volume per unit time or stream height as a function of time) from the following:

1. Recorded climatological data, precipitation, evaporation, and (for snowmelt situations) temperature,

2. Measurable watershed characteristics such as drainage area and friction of the watershed in impervious surfaces, and

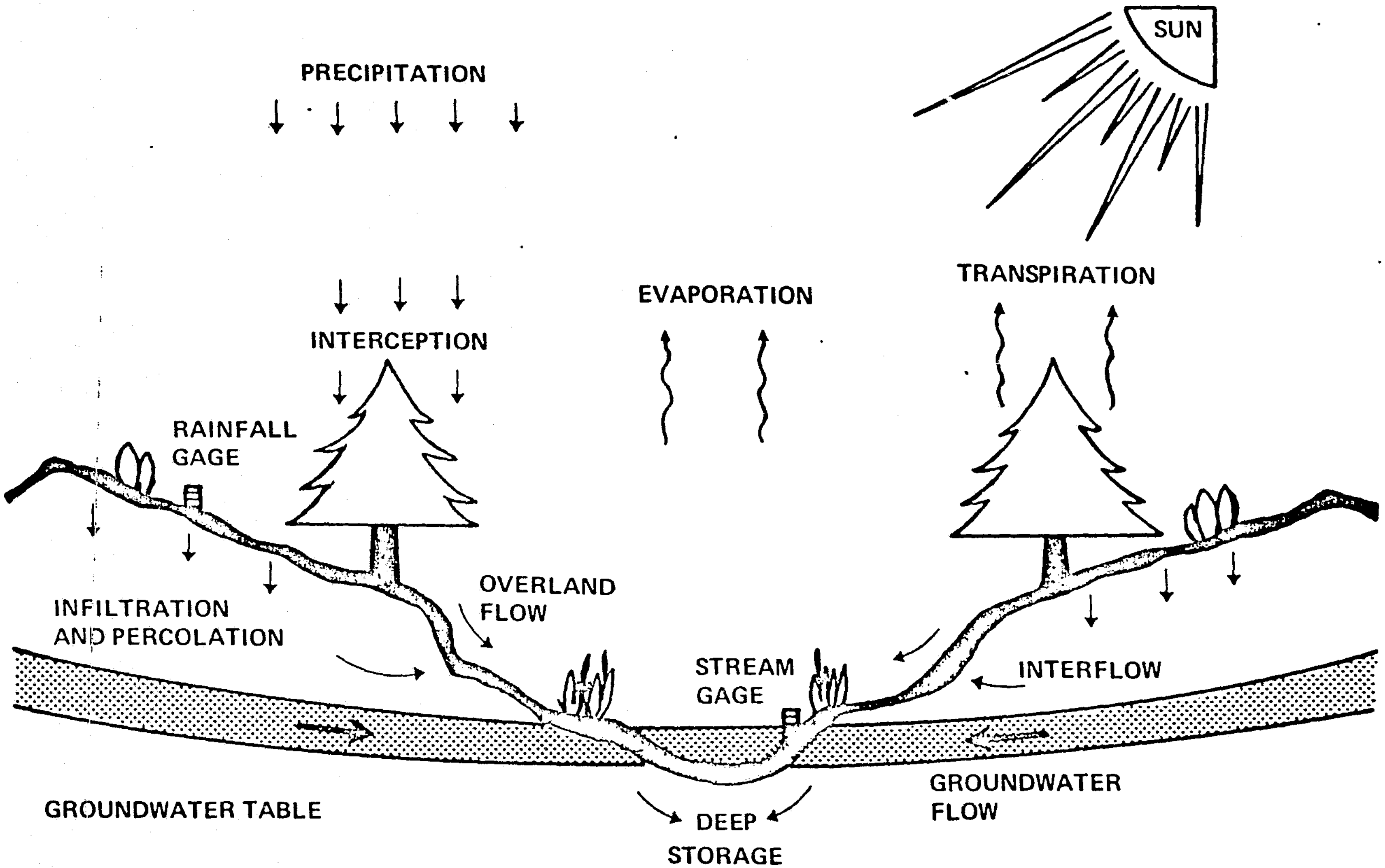


Figure 2-1. Cross Section of Idealized Rural Catchment

3. Parameters used in the computation process which are known to vary in magnitude among watersheds but have not been quantitatively tied to specific measurable watershed properties. For example, one parameter indexes the capacity of the soil of the watershed as a whole to retain water.

The third class of inputs requires a trial-and-error series of calibration runs to quantify a set of model parameters which will synthesize flows with acceptable accuracy.

Figure 2-2 depicts the accounting of moisture entering the watershed until it leaves by streamflow, evapotranspiration, or subsurface outflow. A series of relations, each based on empirical observation or theoretical description of a specific hydrologic process, is used to estimate rates and volumes of moisture movement from one storage category to another, in accordance with current storage states and the calibrated watershed parameters. The model routes channel inflow from the point where it enters a tributary channel to the downstream point for which a hydrographic is required. The flow chart provides the basis for the logical interlocking of a set of empirical equations into the algorithm that is at the heart of the model.

2.3 MODEL PARAMETERS AND CALIBRATION

When a user applies a simulation model to a watershed, there are several parameters whose values he must initially guess and subsequently adjust, between trial runs of the model and comparisons of synthesized with observed flows. This trial-and-error calibration requires ingenuity, understanding of the sensitivity of simulated flows to specific parameter adjustments. The process is aided greatly by a thorough understanding of the hydrologic process and by the guidance published by Crawford and Linsley (1962, 1966). Through careful parameter adjustment, one can cause simulated flows to approximate recorded flows but never to match them exactly. Several combinations of parameter values can produce comparable results from an overall viewpoint, and the final choice may well hinge on whether a particular comparison emphasizes flood peaks, annual runoff volume, or some other hydrographic feature. The final acceptance of a set of parameters may depend heavily on subjective factors.

Figure 2-3 lists the principal inputs (exclusive of control options) used by the watershed model to simulate streamflow. Climatological data can be obtained from precipitation records or can be hypothetical, the latter being useful in generating rainfall-runoff predictions. The inputs classed as "overland flow parameters" and "watershed parameters" are readily obtainable from analysis and interpretation of images (maps and/or photographs). The inputs on the right side of the figure must be estimated by a calibration process.

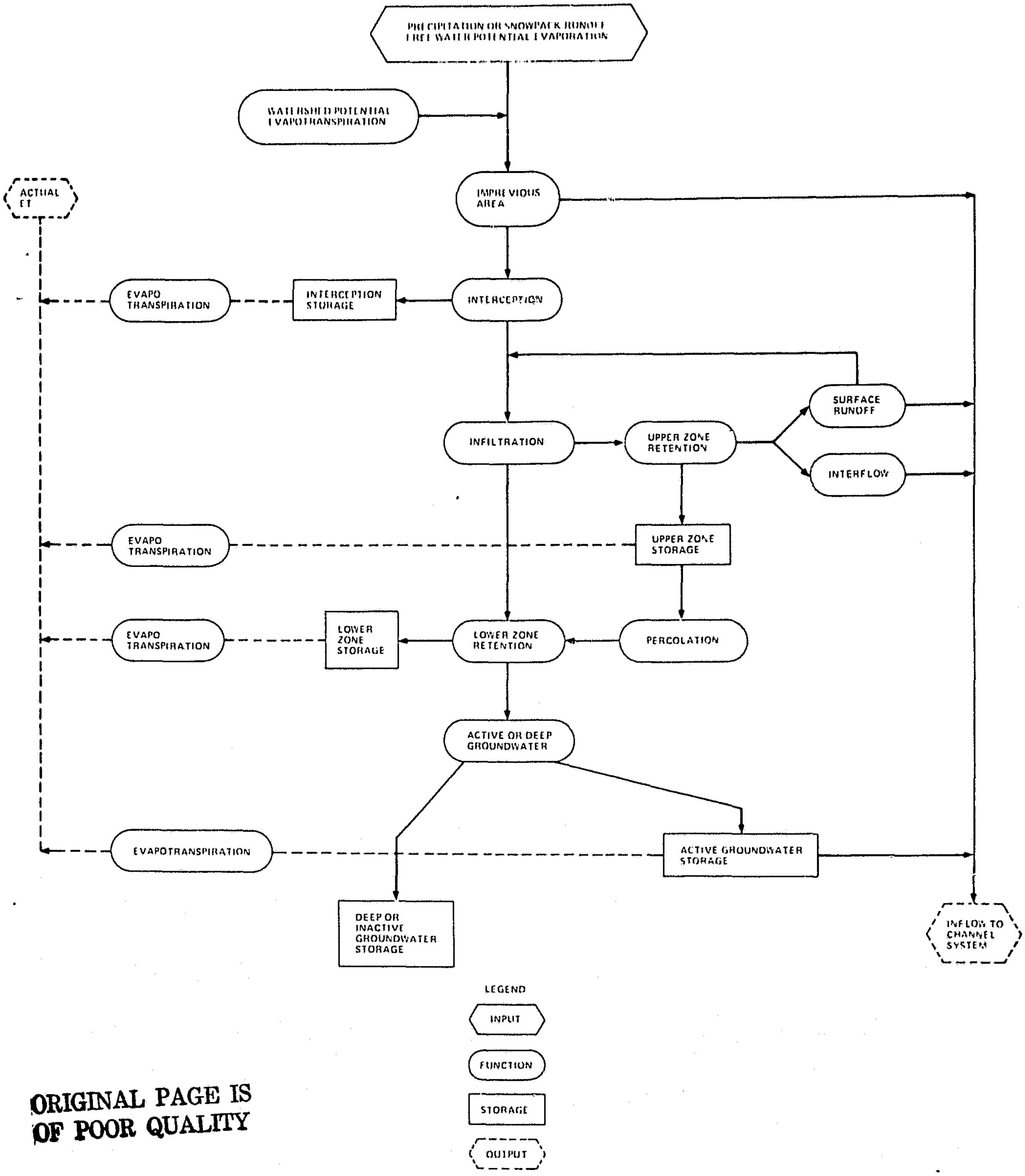
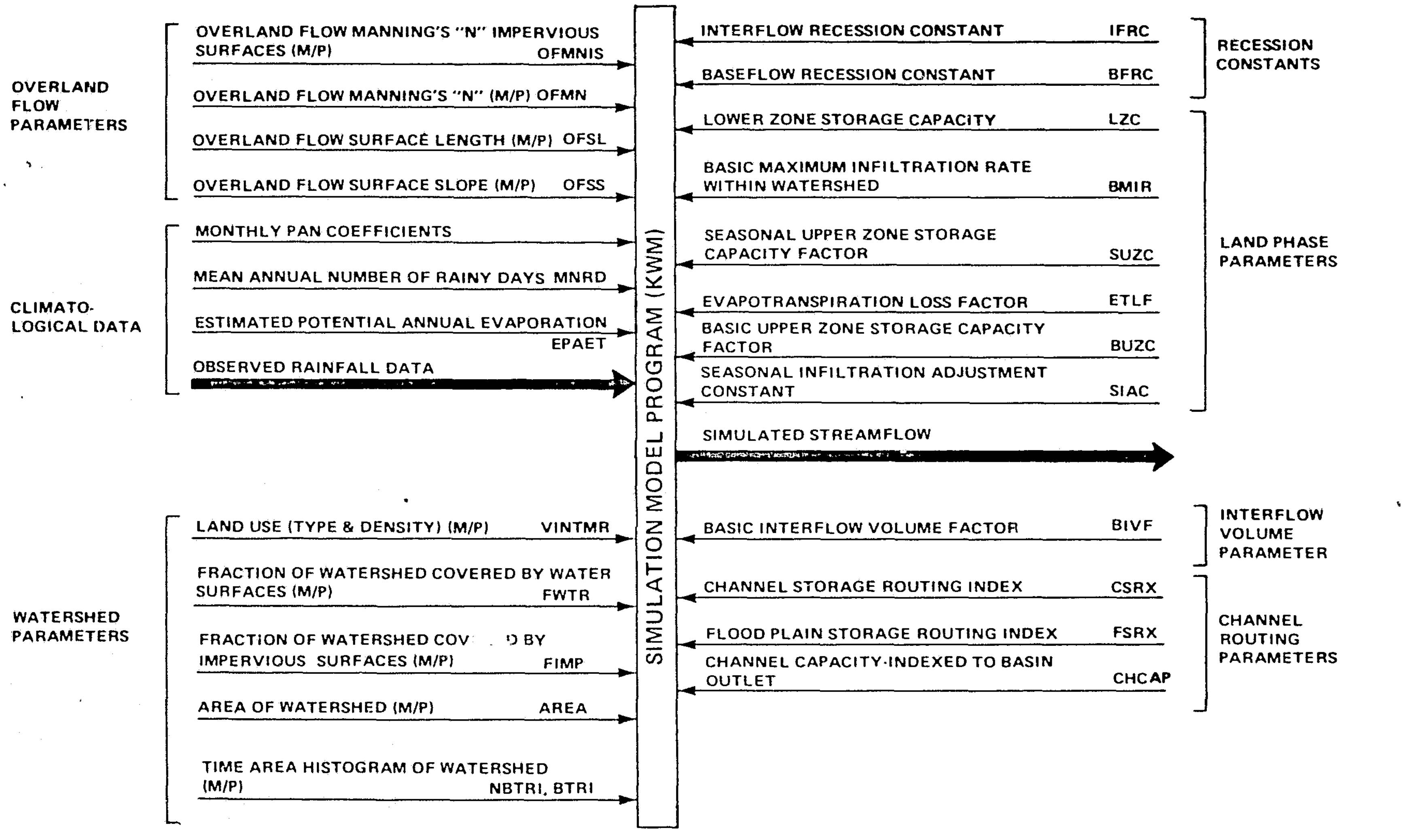


Figure 2-2. Moisture Accounting in the Stanford Watershed Model



NOTE: M/P - MAPS AND/OR PHOTOGRAPHS

Figure 2-3. Simulation Model (KWM) Inputs and Outputs

In developing OPSET, Liou (1970) provided a tool for calibrating the KWM with a minimum of subjective decisions. The parameter optimization concept is depicted in flow chart form in Figure 2-4. The input data consists of control options and initial conditions as well as the inputs listed in Figure 2-5. A simulation is performed, one year at a time. The synthesized flows are compared with the observed flows. An objective function is used to determine when an optimum set of parameters has been found. If the best match has not been achieved, parameters are again adjusted and the simulation run again. This sequence is repeated until a satisfactory parameter set has been quantified.

Figure 2-5 also lists the outputs of OPSET, in addition to simulated streamflow. (Comparison with Figure 2-3 shows the relationship to the simulation model.) These parameters are the most difficult to measure directly and are the ones to which simulated flow values are sensitive. The calibration process should be based on three separate water years for the same basin. Simulation model parameters are then derived by averaging the results of the three calibration runs. A minor modification to OPSET has been implemented to generate a more precise Base Flow Recession Constant (BFRC). As it is presently designed, OPSET estimates parameters which produce accurate simulations of major winter storms (with respect to flood peak magnitude and timing) but misses summer and autumn storm peaks by significant factors. Manual adjustments are required to achieve accurate simulation in the latter. An improvement in OPSET efficiency could be achieved by modifying it to calibrate on the basis of several consecutive years rather than one year at a time.

Snowmelt parameters have not been treated in the foregoing because OPSET does not estimate them. Their quantification is described in Section 4.1.

2.4 INPUT DATA PREPARATION

Simulation of a watershed requires (1) acquisition, formatting and integration of a historical data base, (2) quantification of some of the model parameters from direct observation, measurement and application of empirical relationships, and (3) calibration, the adjustment of the remaining parameters to achieve an acceptable match between simulated (synthesized) and actual streamflow. After calibration, the system may be used to predict streamflow resulting from any given precipitation event.

The historical data base for the system is constructed from the following types of data.

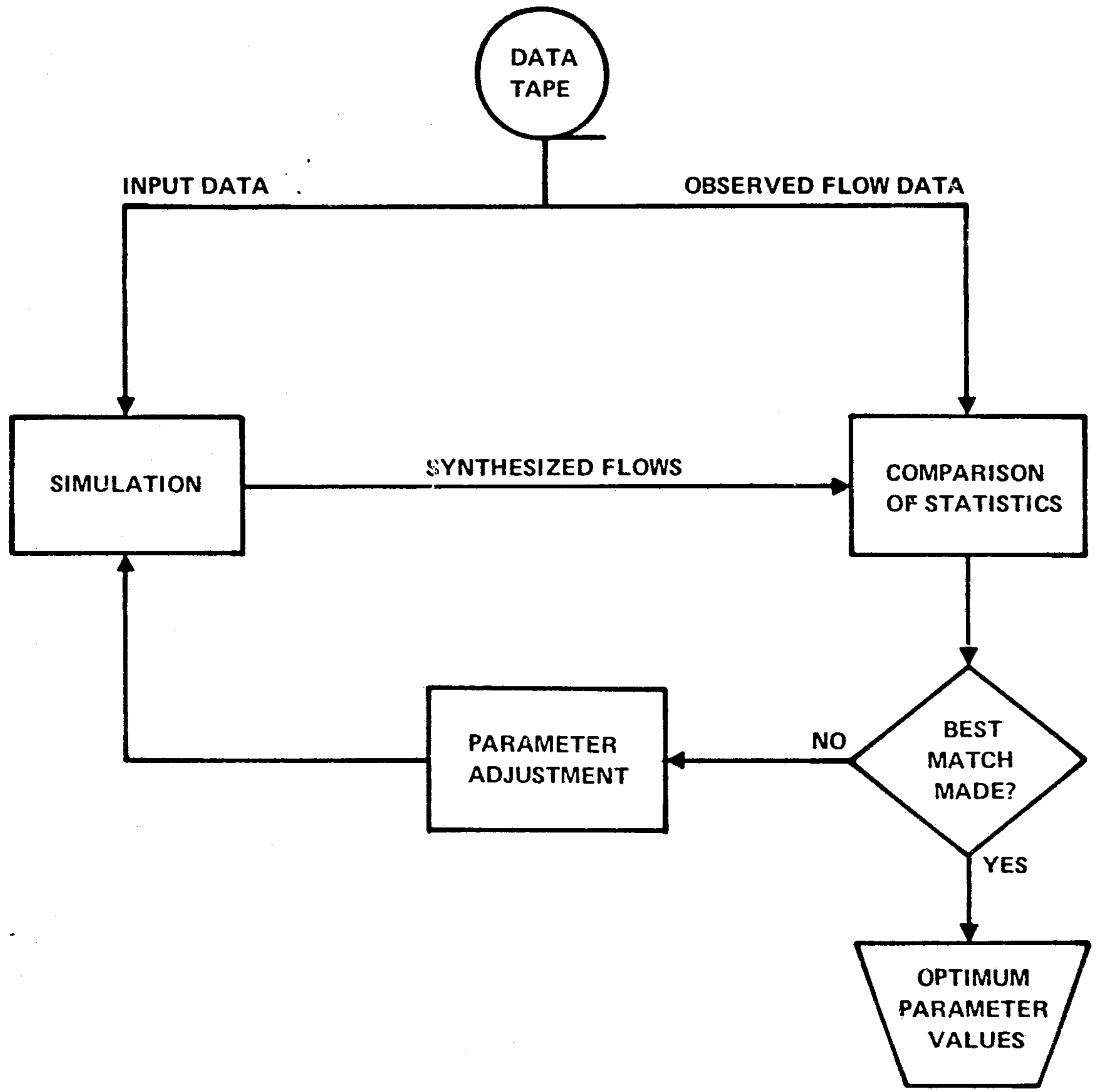
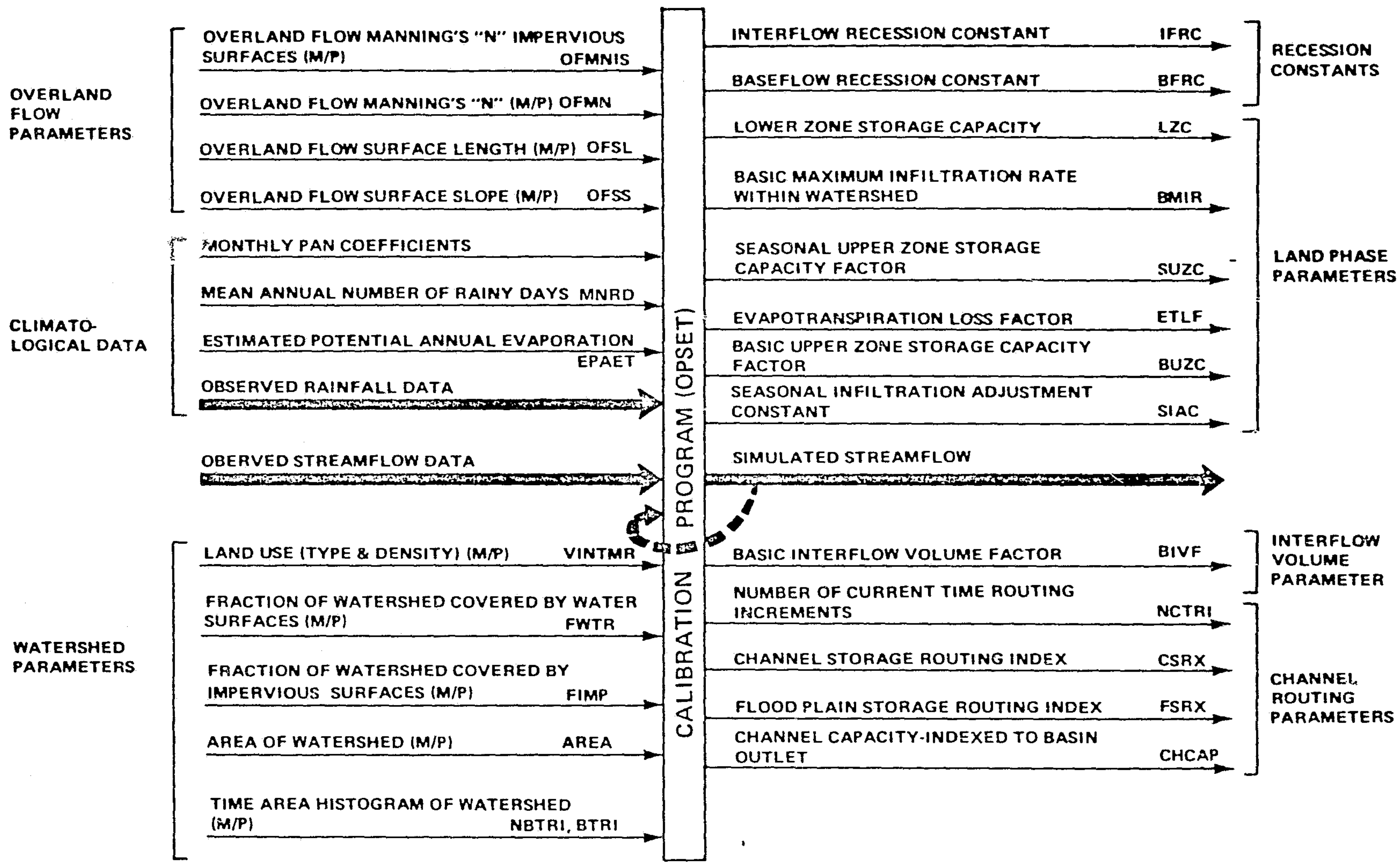


Figure 2-4. Parameter Optimization Concept



NOTE: M/P - MAPS AND/OR PHOTOGRAPHS

Figure 2-5. Calibration Program (OPSET) Inputs and Outputs

- o Precipitation records - hourly and daily
- o Stream stage charts - actual strip chart hydrographs
- o Rating tables for conversion of stream stage charts from height (feet) to flow rate (cubic feet per second, cfs)
- o Daily streamflow (discharge) records
- o Temperature records - maximum and minimum each day (used with snow routine only)
- o Evaporation data - three options:
  - Daily evaporation and monthly pan coefficients; or
  - Total annual evaporation, mean annual number of rainy days and estimated potential annual evapotranspiration; or
  - Average daily evaporation values over ten-day periods through the year and monthly pan coefficients.
- o Snowmelt data arrays

Data are converted from published documents or charts or magnetic tape, to digital formats suitable for input to the calibration and simulation programs, as described in Section 3.

2.5 OPERATION

Once the model parameters and historical data have been integrated into a master data bank, actual operation of the models, whether by batch processing (over-the-counter or remote job entry) or interactive will depend on the configuration, capabilities and procedures of the supporting computer center.

2.6 OUTPUTS

There is a variety of outputs available from the multi-year simulation program in the NASA-IBM system. The operator and analyst can choose those which best suit his needs from the following.

- o A tabulation of hourly synthesized streamflow, with daily values for the following:

- peak flow and time of peak
- snowpack depth
- snow total moisture density
- snow albedo index
- total accumulated negative snowmelt
- snowpack liquid water content

o A table of monthly annual totals.

o A yearly statistical summary.

o A table of mean daily reference streamflow, with monthly and annual totals.

o A table of mean daily simulated streamflow, with monthly and annual totals.

o A table of monthly moisture storages and indices.

o A table of flow duration and error statistics.

o A list of the 20 highest clock hour rainfall events in the water year.

o A list of the 20 highest clock hour overland flow runoff events in the water year.

o A table of daily soil moisture.

o A comparison table of storm events, reference and simulated, with respect to peak flow, time of peak, and runoff, one table per storm event.

o Total daily and monthly statistical summary.

o Print-plots for total year, each month and each storm event.

o Tapes of data for SC-4020 plot outputs for total year, each month and each storm event.

In the forecast and past-run modes, the following outputs are available (see Section 7).

- o Superimposed forecast hydrographs for worst case, forecast and zero precipitation.
- o Superimposed simulated and observed hydrographs for past (fine tune) runs.
- o Tables of hourly streamflow by day for each forecast case.
- o Table of daily precipitation, surface runoff, interflow, baseflow, and stream runoff for each forecast case.
- o Table of peak flow, time of peak, runoff and total precipitation for the forecast period for each forecast case. This table also shows differences and percent differences between worst case and forecast peaks.
- o Table of soil moisture storages at the end of each day of the forecast period.

SECTION 3

BASIC DATA*

The input data required to operate the model consists of a minimum basic core plus supplemental data required by various control options in the program. The various data components will be discussed in the order in which the data is read into the program, which is also essentially the order in which the data is arranged in the data deck.

3.1 TIME-AREA-HISTOGRAM DEFINITION

Both the original Stanford Model IV (Crawford and Linsley, 1966) and the Kentucky Watershed Model (Ross, 1970; Liou, 1970) use an empirical routing procedure developed by C. O. Clark (1945). This procedure routes channel inflow to the basin outlet using a time-delay histogram. The time-area histogram divides the basin into zones of equal travel time by drawing isochrones on a topographic map. The fraction of the watershed area within eash set of isochrones defines the time-area histogram. The procedure for determining these zones is outlined below.

1. Determine the Time of Concentration

   The time of concentration (T_c) is defined as the time required for water falling on the most distant part of the watershed to reach the outlet via the channel. It can be estimated using the empirical equation of Kirpitch (Chow, 1964):

   $$T_c = .0078 \frac{L^{0.77}}{S^{0.385}}$$

   When T_c is in minutes; L is the length in feet of the basin area measured from the channel outlet along the channel, and in a direct line from end of the channel to the furthest point in the basin; and S is the ratio of the length (L) to the fall in feet from the furthest point to the basin outlet.

2. Determine the average velocity of a flood wave through the basin:

   $$V_{av} = \frac{T_c}{L}$$

* This section draws heavily on Striffler (1973) for organization and interpretation.

3. Select a desired isochrone spacing. Since routing is either done on a 15-minute or hourly basis in the model, the isochrone spacing selected should agree with the routing interval. On a large watershed, longer intervals will help hold down computer time requirements.

4. From the average velocity, calculate the distance required for the water to flow one isochrone distance. For example, if the average velocity is 4.0 mph and isochrone interval selected is 15 minutes, an isochrone would cross the channel every mile above the basin outlet.

5. Divide the basin into isochrone areas by measuring the channel distance between isochrones and determining the watershed area contributing flow to that section of the channel. Planimeter the area between each set of isochrones and determine the fraction of basin area in each area. Beginning with the isochrone area above the outlet, these fractions represent the time-area histogram. Data required by the model includes the number of base time routing increments (NBTRI) and the base time routing increment (BTRI) data array (see Figure 3-1).

3.2 SNOWMELT DATA ARRAYS

If snow and snowmelt are important processes on the watershed and called in the model (CONOPT 7 = 1), the following data arrays are required.

FIRR - The fraction of incoming radiation reflected by a snow surface as a function of age.

This is an array of 15 values which is used to adjust snowmelt rates as snow surface albedo changes. It is well known that snow surface albedos change with age and also with rainfall on the surface. Snow albedos have been shown to vary from a maximum of about 0.80 for new fallen snow to a minimum of about 0.40 for a ripe snow pack during the melt season. Under melting conditions, the albedo can change from the maximum to the minimum in about fifteen days (Figure 3-2). This relationship is the basis for the FIRR array. Anderson and Crawford (1964) discussed the application of the reduction factor in the Stanford Snowmelt Model but didn't discuss how they derived the array used. Their data are also plotted on Figure 3-2. In this study, adjustment factors were taken directly from the albedo curve. A more representative array could be determined if data for a total energy budget were available for the watershed.

RICY - Radiation Incidence over the Calendar Year.

The RICY data array is an array of 37 values, each representing an adjustment factor to the snowmelt rate for each 10-day period during the calendar year. In the snowmelt model of Anderson and Crawford (1964), snowmelt is calculated on the basis of a degree day heat input to the snow pack.

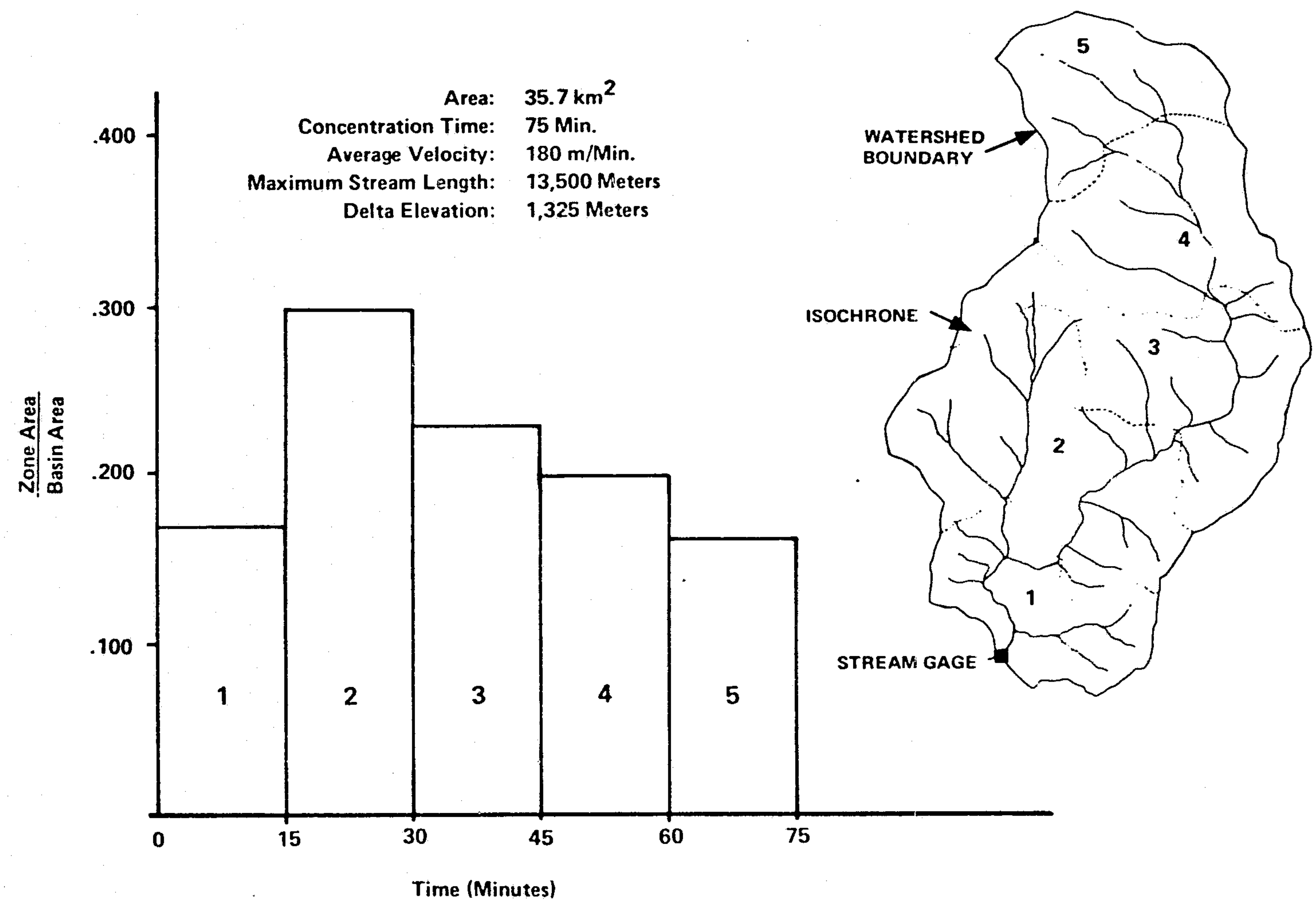


Figure 3-1. Typical Watershed Divided by Isochrones

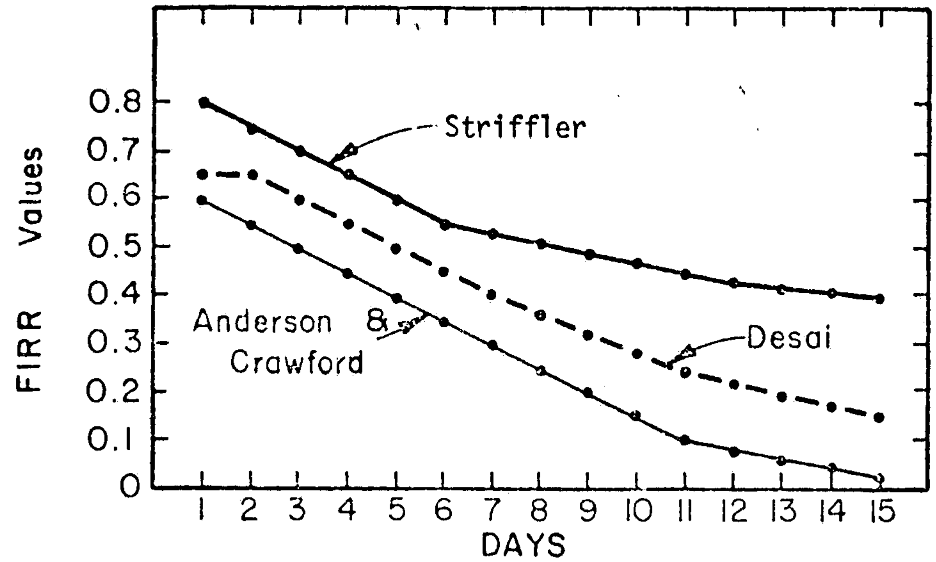


Figure 3-2. FIRR Array

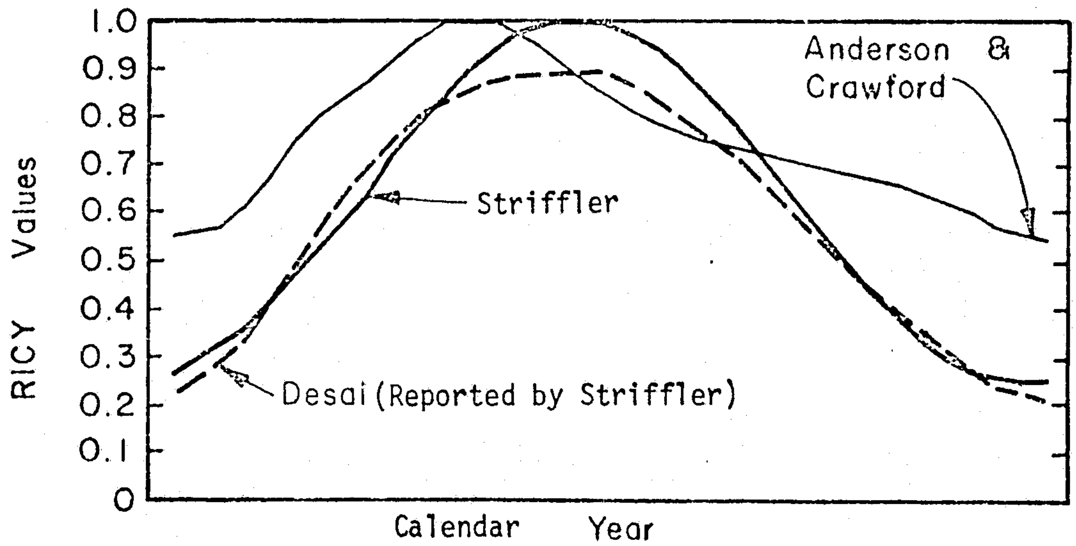


Figure 3-3. RICY Array

The parameter BDDFSM represents the maximum melt rate which would occur during the melt season. However, melt does not occur at the maximum rate during the melt season; and since melt is a function of solar energy as well as air temperature, the maximum melt rate (BDDFSM) represents the maximum input of both sensible heat and solar energy. Anderson (1968) calculated degree day melt factors using daily streamflow hydrographs and a continuous energy balance calculation for the snow pack. His data show a seasonal distribution of melt factors which roughly approximates the distribution of solar energy. Anderson (reported by Striffler) suggests that the RICY data array should approximate a sine wave with the maximum occurring on June 21, and the minimum, about one-fourth the maximum, on December 21. A sine wave distribution and the data array used in the Stanford Snowmelt Model are shown in Figure 3-3. Desai's data was determined from the seasonal distribution of radiation data at Laramie, Wyoming.

DPSE - Dated Potential Snow Evaporation.

In the Stanford Snowmelt Model, evapotranspiration and evaporation from the snow surface are considered separately. The DPSE data array is the data source for snow evaporation and represents daily snow evaporation for 10-day periods during the water year. In the calculations, snow evaporation does not occur if the daily minimum temperature is greater than 32° or if the snow pack total water content is less than the daily potential snow evaporation.

Anderson and Crawford (1964) derived snow evaporation data using the following equation and data from the Central Sierra Snow Laboratory (CSSL) watershed:

$$E = \frac{b'V_W}{Q_t} (\rho_a - \rho_s)$$

where E is evaporation in inches, b' is an empirical constant, V_W is the wind velocity, Q_t is the thermal quality of the snow (percent by weight of ice), ρ_a is the vapor pressure of the air in millibars, and ρ_s is the saturation vapor pressure over the snow.

3.3 EVAPORATION DATA ARRAYS

Three different forms of evaporation data can be used in the model (CONOPT 3 = 0,1,2). These include daily potential evapotranspiration (DPET) values (CONOPT 3 = 0), average daily potential evaporation values for 10-day periods (CONOPT 3 = 1), and an average annual potential evapotranspiration value (EPAET). Although DPET is defined as "dated potential evapotranspiration," the Kentucky Report (Ross, 1970) reduces evaporation data to lake evaporation which is derived from measured pan evaporation and a pan coefficient. Thus DPET values can be measured pan evaporation data or calculated potential ET data. Ideally daily pan evaporation data from a pan located within or adjacent to the watershed should be used. However, since good pan data are rare in most parts of the country and particularly the mountain west, calculated PET values, using an appropriate ET formula will suffice.

In this study, the Hamon Formula (Hamon, 1961) was used:

$$PET = C\ D^2\ Pt$$

in which PET = potential ET in inches, C is a constant, D is the possible hours of sunshine in units of 12 hours, and Pt is the saturated water vapor density (absolute humidity) in grams per cubic meter at the mean daily temperature. The constant C was adjusted so that calculated PET values agreed with a partial pan record in the watershed.

Although daily values of potential ET are most desirable in the daily accounting in the model, averages over 10-day periods can be used if it is desirable to reduce the calculation required to assemble daily values. DPET values for 10-day periods are input as an array of 37 values. The 37 values should correspond to the time distribution shown in Table 3-1.

The third option (EPAET) (CONOPT 3 = 2) requires an annual potential evaporation value plus a correction factor for the number of rainy days during the year (MNRD). These values can either be measured values for a particular year or average values for a number of years. Average values should be used for years in which climatic data are not available. Under this option, an adjusted annual PET value is calculated and distributed over the water year by subroutine EVAPDAY. The EPAET (estimated potential average evapotranspiration) and MNRD (mean number of rainy days) are the only values read, in this option.

If DPET data are read (CONOPT 3 = 0 or 1) an array of monthly pan coefficients (EPCM) must also be entered. This consists of an array of 12 values comprising the evaporation pan coefficient for each month of the water year. If the DPET values are already converted to lake evaporation, or potential evapotranspiration values, pan coefficients should be read as 1.0's.

3.4 STREAMFLOW DATA

Daily discharge data is the average volume in cubic feet of water per second that flows past the stream gage during a 24-hour period. This data exists on magnetic tape and/or written tables for all stream gages in the Tennessee Valley. The data format which exists on magnetic tape must be altered to be compatible with the simulation model. Where the data exists in written tables, it is necessary to manually extract that information, convert to punched card format, and develop a listing compatible with model requirements.

For operation of the OPSET program, it is necessary to select up to five flood hydrographs for each of the years for which the model is to be calibrated. This requires a manual search of precipitation and discharge records to select storms useful to the calibration. The digitized input data include the number of hydrographs chosen and three parameters related

Table 3-1. 10-day Intervals for Averaging Evaporation Data (Ross, 1970)

Oct.	1-Oct. 10	Apr.	1-Apr. 10
Oct.	11-Oct. 20	Apr.	11-Apr. 20
Oct.	21-Oct. 30	Apr.	21-Apr. 30
Oct.	31-Nov. 9	May	1-May 10
Nov.	10-Nov. 19	May	11-May 20
Nov.	20-Nov. 29	May	21-May 30
Nov.	30-Dec. 9	May	31-June 9
Dec.	10-Dec. 19	June	10-June 19
Dec.	20-Dec. 31	June	20-June 29
Jan.	1-Jan. 10	June	30-July 9
Jan.	11-Jan. 20	July	10-July 19
Jan.	21-Jan. 30	July	20-July 29
Jan.	31-Feb. 9	July	30-Aug. 8
Feb.	10-Feb. 19	Aug.	9-Aug. 18
Feb.	20-Mar. 1*	Aug.	19-Aug. 28
Mar.	2-Mar. 11	Aug.	29-Sep. 7
Mar.	12-Mar. 21	Sep.	8-Sep. 17
Mar.	22-Mar. 31	Sep.	18-Sep. 27
		Sep.	28-Sep. 30

* This is an 11-day interval on leap years.

to each hydrograph: day of occurrence of the flood peak, hour of occurrence of the flood peak, and flow rate at the peak. These hydrographs parameters are essential for the OPSET program to determine watershed model routing parameters, so that total flows will represent accurate predictions, with respect to the time of occurrence of hydrograph peaks as well as the total volume of flow for a given period of time. In practice the selected storm hydrograph parameters are not available in daily discharge records. It is necessary to obtain them from the strip charts produced by the stream gage recorders. Rating tables are also digitized and stored for conversion of gage height readings into flow rate.

The procedure employed to obtain this data requires manual analysis of each strip chart and manual recording of the rise and fall of the stream gage on an hourly basis. The time frame should extend from midnight of the day in which the storm occurred until some time at which the stream height returns to or approaches its initial stage. This hourly height recording is then formatted for entry into the computer where a subroutine will fetch the appropriate rating table into memory and convert the data to cubic feet per second. This flood hydrograph data is then in a usable form when required by the simulation model.

3.5 CLIMATIC DATA

Climatic data required in the model includes data required in the basic operation of the model and data required by various optional features of the model. These are described in the order in which they are read into the model.

3.5.1 DMXT, DMNT - DAILY VALUES OF MAXIMUM AND MINIMUM TEMPERATURES

Daily maximum and minimum air temperature is required in the model if the snowmelt subroutine is called (CONOPT 7 = 1). These values are read in as an array of alternating maximum and minimum values for each day of the water year. Since air temperatures vary over a watershed, recorded temperatures from a station (preferably within the watershed) are adjusted by the main program to the mean elevation of the basin. Adjusted temperatures are then used for the remainder of the calculations involving temperature.

3.5.2 HOURLY PRECIPITATION DATA

Hourly precipitation data in digital form is the primary input. In a very small watershed having its own hourly precipitation gage, one can (with reasonable safety) assume that the gage reading applies uniformly to the entire watershed. This assumption (which is implicit in both programs) departs from reality more and more with increase in watershed size. It has been necessary to implement a method whereby several precipitation records are used to synthesize a single hourly rainfall history for each watershed or subwatershed.

The number of precipitation stations associated with any given watershed may vary from one station located twenty or thirty miles from the watershed centroid to five or six stations located within or closely adjacent to the watershed boundaries. Typically, a watershed will have one or two hourly stations, and one or more daily stations. In addition to the varying distances of these stations from the centroid, the reading time for the daily stations might be different. It is also quite likely that data will appear from the several gages in both magnetic tape and tabular formats. The latter must be manually extracted from the tables and converted to punched data card format.

The precipitation gage outputs are assigned weighting factors, using the Thiessen technique (Linsley, et al, 1958; Chow, 1964) in accordance with their physical locations relative to the basin centroid. A software program developed by IBM automatically performs the interpolation and correlation of the precipitation data. This program accepts all precipitation data, the reading time for each daily station, and the weighting factor developed from the Thiessen Analysis, and produces an hourly precipitation record for the applicable water years associated with a given watershed. This hourly precipitation data record is then used as one of the climatological inputs required by the models.

SECTION 4

PARAMETER SELECTION

The operating parameters required for proper model operation include eleven snowmelt parameters, one output parameter, four watershed parameters, ten soil water parameters, five overland flow parameters, six channel routing and ground water parameters, and five starting moisture condition parameters (Table 4-1). The parameter values are used in the model as factors or coefficients in the various process calculations or as starting points in the moisture accounting routines. A number of parameters describe specific watershed characteristics and can be determined from topographic maps, while other parameters serve to control process rates and can be estimated from the literature. Accurate determination of parameter values is desirable from a hydrologic standpoint, since accurate simulation of the various hydrologic processes requires selection of parameter values representative of the particular watershed being tested. The sensitivity of the model to the various parameters varies considerably among the parameters. In some instances, small changes to parameters cause large changes in simulation results, while other parameters can be varied substantially without affecting simulation results significantly. Since in most applications some parameters cannot be determined accurately, the customary procedure is to vary the parameters until the best fit or output simulation results. This calibration procedure can be completely manual or computer aided.

4.1 SNOWMELT PARAMETERS

BDDFSM - The basic degree day factor for snowmelt. Although defined by Liou (1970) as a degree day melt factor, this parameter is actually a degree hour melt factor, since the melt calculation is done hourly throughout the melt season. The parameter value represents the amount of melt which will occur in one hour for every degree F above a base temperature, usually 32°F, during the maximum melt rate season. The maximum melt rate calculated is reduced by several other factors, since it is known that degree day melt factors are not uniform over a melt season (Corps of Engineers, 1956). This parameter is important in determining the timing of snowmelt runoff and the height of peak runoff events during the snowmelt season. It is difficult to determine for any particular watershed. However, the values used by Anderson and Crawford (1964) ranged from .0035 to .0085. This parameter is usually optimized for a best fit value.

SPTWCC - The snow pack minimum total water content for complete basin coverage. In mountain watersheds with large elevation differences, considerable snow accumulation can take place on the upper watershed before the entire watershed is covered. This parameter attempts to define the water content at the point

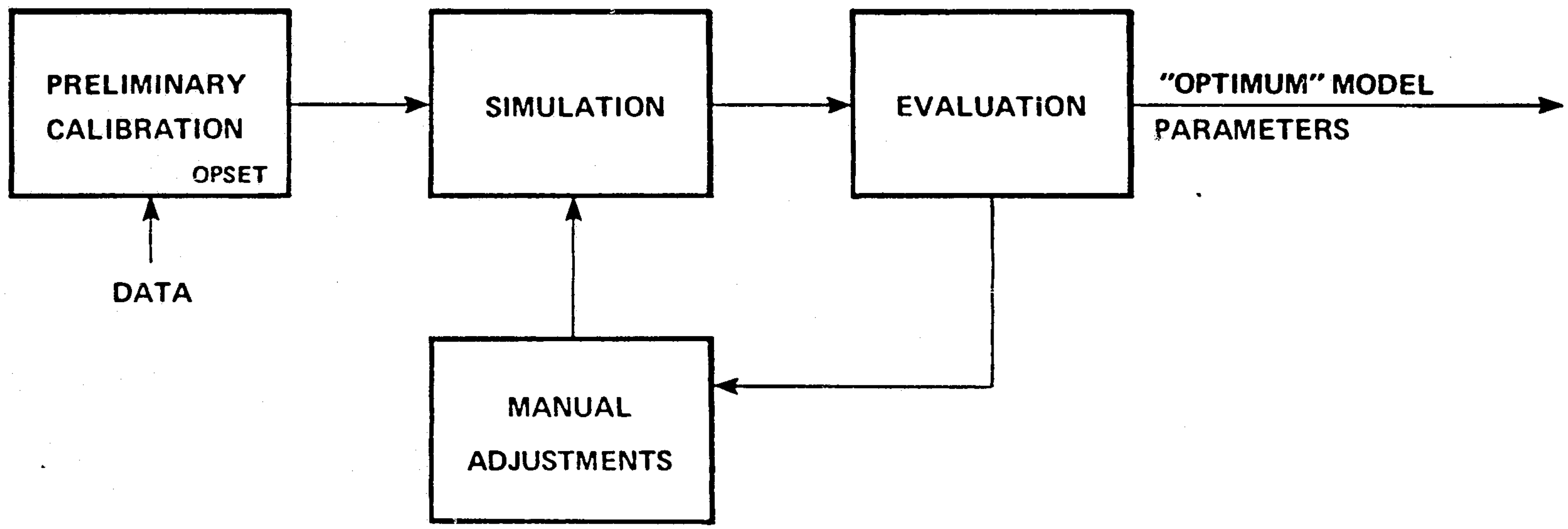


Figure 4-1. Calibration Methodology

Table 4-1 - Parameters Required in the Kentucky Watershed Model

Snowmelt Parameters

BDDFSM - Basic degree day factor for snowmelt.
SPBFLW - Snowpack basic maximum fraction in liquid water.
SPTWCC - Snowpack minimum total water for complete basin cover.
SPM - Snow precipitation multiplier.
ELDIF - Elevation difference between base temperature station and mean basin elevation.
XDNFS - Index density of new fallen snow.
FFOR - Fraction of the watershed forested.
FFSI - Fraction of snow intercepted.
MRNSM - Maximum rate of negative snowmelt (chilling).
DSMGH - Daily snowmelt from ground heat.
PXCSA - Precipitation index for changing snow albedo.

Output Parameter

RMPF - Requested minimum daily peak flow to be printed.

Watershed Parameters

RGPMB - Recording gage precipitation multiplier.
AREA - Area of the watershed.
FIMP - Fraction of watershed covered by impermeable surfaces.
FWTR - Fraction of watershed covered by water surfaces.

Soil Water Parameters

VINTMR - Vegetation Interception, maximum rate.
BUZC - Basic upper zone storage capacity factor.
SUZC - Seasonal upper zone storage capacity factor.
LZC - Lower zone storage capacity.
ETLF - ET loss factor.
SUBWF - Subsurface water flow out of the basin.
GWETF - Ground water evapotranspiration factor.
SIAC - Seasonal infiltration adjustment factor.
BMIR - Basic maximum infiltration rate within basin.
BIVF - Basic interflow volume factor.

Overland Flow and Interflow Parameters

OFSS - Overland flow surface slope.
OFSL - Overland flow surface length.
OFMN - Manning's n for overland flow.
OFMNIS - Manning's n for impervious surface.
IFRC - Interflow recession constant.

Channel Routine and Groundwater Parameters

CSRX - Channel Storage routing index.
FSRX - Flood plain storage routing index.
CHCAP - Channel capacity indexed to basin outlet.
EXQPV - Exponent of flow proportional to velocity.
BFNLR - Base flow nonlinear recession adjustment factor.
BFRC - Base flow recession constant.

Starting Moisture Values as of Oct. 1.

GWS - Current groundwater storage.
UZS - Current upper zone storage.
LZS - Current lower zone storage.
BFNX - Current value of base flow recession index.
IFS - Current interflow storage.

the entire basin becomes covered with snow. The parameter is used to adjust snowmelt for incomplete snow cover on the basin. When the water equivalent is less than the parameter index value, it is assumed that the snow covered area is proportionately less and melt is adjusted accordingly. This parameter is generally estimated and optimized for a best fit. A knowledge of snow pack water contents and accumulation patterns on the watershed is essential to assign a realistic value to the parameter.

SPM - The snow precipitation multiplier. If simulated snowmelt runoff is consistently lower than measured runoff, it is reasonable to assume that the precipitation gages are not as efficient in measuring snow catch as rain. This occurs frequently for gages in exposed sites or non-shielded gages. This parameter attempts to adjust for this difference by increasing the hourly precipitation value for all snowfall events. This parameter is estimated and optimized for a best fit.

ELDIF - The elevation difference between the base thermometer and the mean basin elevation. This parameter provides an elevation adjustment for temperature data. Since temperature stations on mountain watersheds are generally located at more accessible sites, usually low on the watershed, a temperature adjustment to estimate mean basin temperature from measured station temperature is required. The parameter, expressed in thousands of feet, is easily determined from a topographic map of the watershed. The parameter is positive if the station is below the mean basin elevation.

XDNFS - Index density of new-fallen snow. Since the snowmelt subroutine maintains a running account of snow depth, it is necessary to know the density of new snow accumulation to the snow pack. New snow density values ranging from 0.004 to 0.34 have been reported (Gray, 1970). Anderson and Crawford (1964) used a density value of 0.05 based on CSSL data (Corps of Engineers, 1956).

FFOR - Fraction of watershed forested. This parameter, called Net Forest Cover by Anderson and Crawford (1964) is defined as the area of forest cover times the average cover density. Thus, a knowledge of the proportion of forest cover plus the average density of the forest is required to estimate this parameter. The proportion of forest cover on a watershed can easily be determined from cover type maps or aerial photographs. For larger watersheds, satisfactory estimates can be obtained from the USGS topographic quandrangle maps with the green forest cover overprint. An estimate of the average canopy density on the watershed is more difficult, although satisfactory estimates can be obtained from cover type maps and aerial photographs. This parameter is important in that it determines snow interception losses and snowmelt adjustments due to reflected radiation.

FFSI - Fraction of snow intercepted. This parameter is defined as the proportion of snow which would be intercepted by a forest canopy if the cover density were 100 percent. This assumes that the amount of snow intercepted is a linear function of the total snowfall. This may be true to a

point but for very large snowfalls or snowfall events accompanied by strong winds, the assumption does not hold. Leaf and Brink (1973a) calculated snow interception of spruce-fir and lodgepole pine forests as 0.15 and 0.10 respectively of the snowfall for canopy densities up to a maximum of 0.30 and 0.20. Anderson and Crawford (1964) use a value of 0.15 for sub-alpine watershed in the Sierras.

MRNSM - The maximum rate of negative snowmelt (chilling). This parameter represents the rate at which liquid water in the snow pack will refreeze when the air temperature falls below freezing. Since the parameter as defined is used empirically to calculate chilling and negative melt, it is best determined by estimating an initial value and optimizing for a best fit. Anderson and Crawford (1964) used values ranging from 0.006 to 0.016 for the various watersheds used in their report. The parameter is important in that it controls the rate and timing of snowmelt. Increasing the value will delay the snowmelt hydrograph.

DSMGH - The rate of daily snowmelt from ground heat. This parameter is an estimate of the amount of water released from the snow pack to the soil from ground heat above. The subroutine assumes that this is a constant rate throughout the season, although in fact it will vary depending upon the depth of the snow pack and the time of year. In those areas where the snow pack is not deep enough to prevent soil freezing, the parameter would be zero. Literature values cited for this parameter are in the order of 0.02 inches per day. Anderson and Crawford used values between 0.0 and 0.02. A value of 0.01 or 0.02 will maintain soil moisture contents and help to maintain a base flow during the winter months.

PXCSA - Precipitation Index for changing snow albedo (inches). In the snowmelt subroutine the maximum snowmelt rate as determined by the basic degree hour factor is adjusted for shortwave radiation inputs and changes in the albedo of the snow pack. Various processes will affect the albedo of the snowpack including aging and the deposition of fresh snow or rain on the pack. The deposition of fresh snow deposit on the snow pack will increase the albedo, while a rain on a snow pack will decrease the albedo. In the accounting procedure, the snowmelt adjustment factor (FIRR) is increased or decreased whenever an index value of new snow or rain is reached. The parameter, PXCSA, represents the index value for determining when changes in the albedo occur. Since the parameter is empirical and does not necessarily represent actual conditions, it is best determined by estimating and optimizing for a best fit. Anderson and Crawford used an index value of 0.2 inches.

4.2 OUTPUT PARAMETER

RMPF - Requested minimum daily peak flow to be printed. This parameter simply controls the printed output of storm hydrographs by specifying a minimum discharge level below which storm hydrographs will not be printed. The parameter does not enter into any one of the calculations and is used primarily to limit paper output.

4.3 WATERSHED PARAMETERS

RGPMB - Recording gage precipitation multiplier - basic. This parameter permits adjustment of the precipitation data. If, during the calibration procedure it is observed that simulated stream discharge values are consistently high or low, it may be due to a consistent under or over estimation of precipitation gage data which may be corrected by this parameter. The parameter, then, represents a ratio between average basin precipitation and precipitation at the recording rain gage station.

AREA - Area of the watershed (square miles). The area of the watershed may be determined from aerial photographs or good quality topographic maps such as the USGS 1:12500 quandrangle maps. For gaged watersheds, the area may be taken directly from the water supply papers.

FIMP - Fraction of impervious surface on the watershed. This parameter refers to the proportion of watershed area which may be considered as impervious to water intake and contributes its runoff directly to channel flow. For most rural or mountain watersheds, this factor will be near zero unless large areas of rock outcrops occur. For urban watersheds, the parameter may be much greater. The parameter is difficult to determine, although aerial photographs and cover type maps may be helpful. A good knowledge of the character of the watershed is also helpful.

FWTR - Fraction of watershed covered by water surface. This parameter refers to the proportion of water surface on the watershed including lakes, ponds, and the area of stream surface. In the calculation it is used to determine the proportion of the watershed at which evaporation occurs at the potential rate. It may be estimated using aerial photographs or maps. Ricci (1972) suggests a technique from Linsley et al. (1949) to estimate the surface area of a channel system as follows:

$$A = \frac{1}{2} BL$$

in which A = water surface area
B = Channel width at outlet
L = total channel length.

4.4 SOIL WATER PARAMETERS

VINTMR - The maximum volume of interception by vegetation (inches). This parameter refers to the maximum volume of water which will be caught and held by a vegetation canopy during a period of rainfall. Ricci (1972) and Ross (1970) both use the table of values suggested by Crawford and Linsley (1966).

Table 4-2. Interception Values for Various Cover Types

Watershed Cover	VINTMR (inches)
Grassland	0.10
Moderate Forest Cover	0.15
Heavy Forest Cover	0.20

Ross (1970) indicates that the parameter was not particularly sensitive and therefore was not optimized. Sensitivity analyses performed by IBM in recent studies support this conclusion.

BUZC - Basic Upper Zone Storage Capacity Factor. This parameter is an index for estimating the storage capacity of the soil surface to store water in depression storage. The parameter is usually estimated initially and optimized for a best fit. Ross (1970) found best fit values ranging from 0.43 to 7.42 for 17 rural watersheds. He also observed a probable relationship between BUZC and the average overland slope, forest cover, and permeability of the "A" Horizon. Clarke (1968) suggested initial values of BUZC ranging from 0.10 to 1.65. Other values reported range from 0.5 to 3.0 Ricci (1972). Sensitivity studies by Ricci (1972) and Ross (1970) indicate that BUZC is not a sensitive parameter with respect to simulated runoff and is used primarily for fine tuning the model.

Crawford and Linsley (1966) suggested a relationship between BUZC and LZC for various slope and cover conditions (Table 3.3).

SUZC - Seasonal upper zone storage capacity factor. The seasonal upper zone storage capacity factor is essentially an index to adjust the BUZC index for seasonal variation. For example, growth of ground vegetation

Table 4-3. Approximate Relationship Between BUZC and LCZ (Crawford and Linsley, 1966)

Surface Characteristics	BUZC
Steep slopes Limited vegetation Low depression storage	0.06 LZC
Moderate slopes Moderate vegetation Moderate depression storage	0.08 LZC
Heavy vegetation or forest cover Soil subject to cracking Mild slopes High depression storage	0.14 LZC

or cultivation of fields may increase the upper zone storage during the summer season. SUZC, like BUZC, is estimated and optimized for a best fit. Ross (1970) reported values ranging from 0.35 to 6.63 for 17 rural watersheds, and observed that higher values of SUZC were associated with more porous soil types. Sensitivity studies by Ricci (1972) indicate that increasing SUZC values will decrease simulated summer storm peaks and runoff volumes.

LZC - Lower zone storage capacity (inches). This parameter is essentially an estimate of the average water holding capacity of the soil. Ross (1970) found a close correlation between LZC and the plant available water capacity. Ricci (1972) defined LZC as the volume of water which may be stored in the soil between the ground surface and the water table, including water draining to ground water. LZC is therefore closely related to the texture of the soil and the depth of the soil mantle. Clarke (1968) suggests that LZC may be estimated as 20 percent of the soil depth. Crawford and Linsley (1966) suggest the following criteria:

For Seasonal Rainfall

$$LZC = 4 + 0.5 \text{ (Mean Annual Rainfall)}$$

For Uniform Rainfall

$$LZC = 4 + .0125 \text{ (Mean Annual Rainfall)}$$

A knowledge of the water holding capacity of the soils in a watershed should be of considerable help in estimating LZC. Leaf and Brink (1973b) use a water holding capacity of 5.3 inches for a sub-alpine watershed in Central Colorado. LZC is a highly sensitive parameter in simulating streamflow. Increasing LZC degrees volume of flow, storm peaks, and snowmelt hydrograph peaks.

ETLF - Evapotranspiration loss factor. The ETLF is described as an index used to determine the maximum evapotranspiration (ET) opportunity on the watershed. The evapotranspiration opporutnity is described as contolling ET losses from the lower zone storage (LZS). A number of assumptions are involved in calculating ET from lower zone storage. First, it is assumed that, on a given day, the ET opportunity will vary over the watershed from zero to some maximum value. The distribution is assumed linear. The maximum rate is determined from the equation:

$$\text{Maximum Rate} = ETLF * LZS/LZC$$

Thus, for drier soil condition, the maximum rate or maximum ET opportunity will be low; but when the soil is at field capacity, the opportunity will equal ETLF.

Crawford and Linsley (1966) suggested initial ETLF values for the following watershed covers.

Table 4-4. ETLF Value for Various Cover Types

Cover Type	ETLF
Barren Ground	0.20
Grassland	0.23
Light Forest	0.28
Heavy Forest	0.30

However, Ross found optimum values ranging from 0.78 to 0.10 for his seventeen rural watersheds, although only two were greater than 0.35. He found a relationship between ETLF and the overland flow surface slope (OFSS) and the presence or absence of forest cover on the flood plain. In general, ETLF decreased sharply with an increase in slope.

Ricci (1972) indicates that yields are sensitive to changes in the parameter and storm peaks are moderately sensitive. Increasing ETLF reduces both yields and storm peaks.

SUBWF - Subsurface water flow out of the basin. This parameter refers to the fraction of water entering ground water storage and leaving the basin through the subsurface flow bypassing the stream gage. Ross (1970) suggests this parameter be estimated and optimized for a best fit. The parameter is usually set at zero unless there is strong evidence that ground water losses occur. A knowledge of the geology of the basin will be helpful in determining potential losses.

GWETF - Ground water evapotranspiration factor. This parameter provides an adjustment to basic ET rates to account for direct plan use of water from ground water. Examples would be watersheds with significant areas of phreatophytes or swamps. The parameter is the fraction of the watershed area covered with swamp or wetland vegetation. This factor is generally zero for most watersheds. This parameter would probably have the same effect as ETLF, decreased yields and peak flows, as the parameter is increased.

SIAC - Seasonal infiltration adjustment constant. This parameter is an index to adjust infiltration rates for seasonal variation. In general, summer infiltration rates are higher than winter rates. Ross (1970) determined a relationship between SIAC and the percent of forest cover on the watershed which he suggests as a guide to determine SIAC values.

Table 4-5 Estimates of SIAC by Percent Forest Cover (From Ross, 1970).

SIAC	Percent Forested Area
.15	0 - 20
.50	21 - 60
.70	61 -100

Values reported by Ross (1970) for his 17 rural watersheds ranged from 0.03 to 3.63. Values reported by Ricci (1972) for a number of other studies ranged from 0.2 to 4.0. It is suggested that an initial value be selected and the parameter optimized for a best fit.

BMIR - Basic maximum infiltration rate. The Stanford Watershed Model infiltration calculation is based on the assumption that infiltration rates within a watershed vary considerably and that this range of rates is distributed linearly from zero to some maximum value at some point on the watershed. The parameter BMIR is used to determine this maximum rate. Crawford and Linsley (1966) suggest values ranging from 0.3 to 1.2. Clarke (1968) suggests that the parameter is approximately twenty percent of the soil permeability.[1] Values reported by Ross ranged from 0.27 to 14.29. Values reported by Ricci (1970) for a number of other studies ranged from 0.65 to 3.25.

The parameter is important since it determines the basic division between surface runoff and infiltration to interflow, soil water, and ground water. As such, it is highly sensitive in the model. Increasing BMIR will cause a decrease in yield and peak flow, and an increase in interflow recession and base flow recession.

BIVF - Basic interflow volume factor. This parameter is an index which controls the time distribution and volume of water entering interflow. It is assumed that, at any point in time, there is a maximum infiltration capacity at some point on the watershed and that the infiltration capacity is distributed linearly over the watershed from the maximum rate to zero. The maximum rate is a function of soil storage capacity, current infiltration rates, and current interflow rate. The current interflow volume multiplier (CIVM) is defined as the product of BIVF and soil storage (LZS/LZC). Increasing the parameter, BIVF, thus increases the portion of water going into interflow. This decreases peak flows and increases base flows or recession flows. BIVF is difficult to estimate. Crawford and Linsley (1966) indicate that the value ranges from 0.5 to 3.0 and is adjusted on a trial-and-error basis. Ross (1970) using optimization procedures, found BIVF values ranging from 0.0 to 3.91. Of his 17 rural watersheds, 14 had no evidence of interflow which was attributed to the small size of his watersheds. Ricci (1972) reported values ranging from 0.65 to 4.5 from a number of other studies. His studies indicated that BIVF is not a sensitive parameter with respect to simulating runoff. However, it appears to be important in adjusting peak flows and recession rates in Rocky Mountain Watersheds where interflow volumes are a significant proportion of the total yield.

4.5 OVERLAND AND INTERFLOW PARAMETERS

OFSS - Overland Flow Surface Slope. This parameter is defined as the average slope of the overland flow surface (inches per foot) and is usually determined from topographic maps of the watershed. Ross (1970) suggests randomly selecting points on the watershed and determining the slope at those points. In this study a one-half mile grid was superimposed on the watershed and the average slope of the overland flow path passing through each point was determined by measuring the length of the flow path and the elevation change along the flow path. The average of these values was taken as the overland flow surface slope.

[1] Ross found that BMIR = 2.3595 ρ_A where ρ_A is the permeability of the A horizon.

OFSL - Overland flow surface length. The average length of the overland flow path (feet) is taken as the average length of the flow surface from a divide perpendicular to the contour to the nearest channel. This parameter is also measured on topographic maps. In this study, the length of the flow distances passing through each of the grid points was measured and averaged to obtain the parameter. A degree of judgement is required to determine whether topographic depression represents channels or not.

In this study neither the OFSS or OFSL parameters are considered critical since, in sub-alpine watersheds, overload flow is considered a rare event. However, in the interest of simulating the hydrologic performance of the basin, the parameter should be measured as accurately as possible.

A sensitivity study by Ross (1970) in which OFSL was changed from 100 to 1,000 feet, increased infiltration opportunity, and slightly increased baseflow and interflow and decreased runoff changes were slight.

OFMN - Manning's n for overland flow. Manning's n for overland flow is used in the overland flow calculation to determine the equilibrium flow depth and the rate of discharge from overland flow surfaces. The roughness coefficient n is generally taken from published tables of n values. Ross (1970) presented a table of n values ranging from .013 for smooth asphalt to .100 for heavy timber. These do not agree with values given by Crawford and Linsley (1966) and Ricci (1972) who give values ranging from 0.012 for smooth asphalt to .400 for forest litter and dense shrubbery. The original table from Crawford and Linsley (1966) is presented here.

Table 3.6 Manning's Roughness Value for Overland Flow for Various Surface Types (From Crawford and Linsley, 1966).

Watershed Surface	Manning's n
Smooth Asphalt	0.012
Asphalt or Concrete Paving	0.014
Packed Clay	0.03
Light Turf	0.20
Dense Turf	0.35
Dense Shrubbery and Forest Litter	0.40

OFMNIS - Manning's roughness coefficient for overland flow from impervious surfaces. This parameter is estimated as discussed above and may be determined using the same table.

IFRC - Interflow recession constant. This parameter controls the rate at which water moves through the upper soil zones. It can basically

be defined as the ratio of the interflow discharge on any day to the interflow discharge of the previous day. Various methods of estimating are available. Barnes (1940) suggests a graphical hydrograph analysis method. Ross (1970) uses a least squares method of estimating IFRC (James and Thompson, 1970) and reported values ranging from .100 to .403. However, he indicated that interflow was a minor flow component on his watersheds and that a minimum value of .400 is required to produce interflow in the model. Other literature values reported by Ricci (1972) range from .001 to .82. The parameter is of slight importance on small watersheds where interflow volumes are small but considerably more important on large mountain watersheds with large snowmelt components.

If recommended procedures for determining the parameter are not used, it can be initially estimated and optimized for a best fit.

4.6 CHANNEL ROUTING AND GROUND WATER PARAMETERS

CSRX - Channel storage routing index. CSRX is a streamflow routing parameter used to account for channel storage effects when channel storage is less than one-half of capacity (CHCAP). In the calculations, the channel system is considered as a reservoir with an outflow and inflow, and temporary storage occurring throughout the length of the channel:

$$Q_2 = \bar{I} - \text{CSRX}\,(\bar{I} - Q_1)$$

where Q_2 is the outflow at the end of a time interval, Q_1 is the outflow at the beginning of the time interval and I is the average inflow during the interval. For a channel with no inflow during the interval, CSRX is essentially equal to the recession rate for the interval.

CSRX can be determined using graphical methods or the equation (Ricci, 1972):

$$\text{CSRX} = \frac{K - 0.5t}{K + 0.5t}$$

Where t is the routing period and

$$K = -Q/\frac{dQ}{dt}$$

Where dQ/dt is the slope of a line tangent to the hydrograph at the point of contraflexure, and Q is the rate of surface runoff at the point of contraflexure.

Ross (1970) reported CSRX values ranging from 0.887 to 0.962 for his 17 rural watersheds. It is satisfactory to estimate the parameter and optimize for a best fit. Increasing the value of CSRX will decrease small flood peaks.

FSRX - Flood storage routing index. FSRX is a streamflow routing parameter to adjust for channel storage plus flood-plain storage when streamflows are greater than twice the Channel Capacity (CHCAP). When large flows are to be routed, the program substitutes FSRX for CSRX. For streamflow values between one-half CHCAP and two CHCAP, the program interpolates values between FSRX and CSRX. Increasing FSRX decreases large flood peaks. FSRX is determined following the same procedure as outlined above for CSRX. Values for FSRX can also be estimated and optimized for a best fit. However, it is necessary to optimize on data which includes flood flows exceeding twice the channel capacity.

CHCAP - Channel capacity indexed to the basin outlet. CHCAP is defined as the flow at the mouth of the channel (gaging station) which is associated with wide spread flooding occurring on the tributary channels. It is not necessarily the flow at which the channel at the gaging station begins to overflow. In the calculation it is used as a reference base to determine at which point to make adjustments in the channel storage volume.

CHCAP can be determined from a hydraulic analysis of the channel dimensions (cross section and profile). Ross (1970) also suggests estimating the gage height at a bank full flow from a topographic map. For streams with a USGS gaging station at the simulation point, a base flooding is usually listed in the USGS water supply papers reporting streamflow at the station. This base value, which is used by the USGS in determining a "base" for reporting flood flows, can be used directly in CHCAP.

EXQPV - Exponent of flow proportional to velocity. This parameter is an exponent used in control option 13 (subroutine RTVARY). Its primary function is to cause flood flows to move downstream faster with the result that storm peaks are magnified. Ross (1970) indicates that Leopold and Maddock (1953) recommend a value of 0.1 for EXQPV but that 0.25 seems to work better.

BFNLR - Base flow nonlinear recession adjustment factor. Some watersheds demonstrate a non-linear ground water recession rate. This may be determined by plotting the ground water depletion hydrograph on a semi-log graph. If the depletion rate is linear, a straight line plot will result. BFNLR is a parameter which permits adjustment to the base flow recession coefficient where a nonlinear depletion is encountered.

The parameter is generally estimated on a trial and error basis. Ross (1970) and Liou (1970) suggest setting the initial value of BFNLR at 1.0 and testing the model to see whether additional adjustments are required. Liou (1970) suggests a method of determining nonlinearity in the base flow recession curve using the distribution of BFRC values calculated for different time intervals

(flow regimes). Ross (1970) suggests that BFNLR values should seldom be dropped below 0.90 although values as low as 0.36 have been reported in the literature (Ricci, 1972).

BFRC - Base flow recession constant. BFRC determines the rate at which base flow recedes as ground water supplies decrease. The parameter may be defined as the ratio of base flow at the end of a twenty-four hour period to the base flow at the beginning of the period. BFRC may be determined using the graphical technique of Barnes (1940). Ross (1970) and Liou (1970) in the OPSET version of the Kentucky Watershed Model, use a least square method of calculating the parameter (James and Thompson, 1970). For watersheds with stratified aquifers supplying water to the stream at different rates, a method of multiple recession constants has been developed by Owen (1970). In SDOPTM, where BFRC is optimized, it is possible to estimate an initial parameter value and optimize for a best fit.

Sensitivity studies by Briggs (1969) indicates that BFRC is an important parameter in simulating peak flows, interflow recession, and base flow recessions (Ricci, 1972). Increasing the parameter value decreases peak flows and interflow recessions and increases base flow recessions.

4.7 STARTING MOISTURE VALUES AT THE BEGINNING OF THE FORECAST PERIOD

GWS - Current ground water storage (inches). This parameter sets the beginning value of ground water storage for the moisture accounting routine. A knowledge of ground water conditions on the watershed at the beginning of the year should be helpful in determining this value. However, since this type of information is not available on many watersheds, particularly high mountain watersheds, it is usually necessary to estimate a value and adjust it in subsequent runs until it "fits." A knowledge of the rainfall/runoff pattern for the month preceding October 1 should also be helpful in setting initial values. Liou (1970, p. 41) outlines a procedure for estimating GWS, based on the recorded October 1 streamflow.

UZS - Current upper zone storage (inches). UZS is the beginning value of upper zone water storage which includes water intercepted on the surface and/or held in depression storage. This value is generally set at zero unless a storm has occurred in the day or two preceding October 1. The value selected should fall between BUZC and zero depending upon the time since the last rain and the amount of rain in the last storm.

LZS - Current lower zone storage (inches). LZS refers to the beginning value of soil water storage and should be some value between LZC and zero. Again, considering the rainfall pattern for a week or two prior to October 1 should be helpful in estimating LZS. If a rain large enough to recharge part of the soil profile has occurred, this should be reflected in the initial estimate. Liou (1970) indicates that LZS is a certain proportion of LZC and that changes

in LZC during optimization should be accompanied by proportional changes in LZS. Liou also presents an empirical technique for estimating beginning values of LZS.

BFNX - Current value of the ground water slope index. Ross (1970) and Liou (1970) define this parameter as the current value of the Base Flow Nonlinear Recession Index. However, the definition of Crawford and Linsley (1966) and Ricci (1972) as the slope of the ground water seem to describe the parameter more appropriately. This parameter is multiplied by the Base Flow Nonlinear Recession Adjustment Factor in the calculation to provide the adjustment for variable recession rates. Ross (1970) suggests an initial approximation of the value is to set BFNX equal to GWS. Briggs (1969) suggests that initial values should be set between 0.15 and 0.25. Other values reported have ranged up to 1.56 (Ricci, 1972).

IFS - Current value of interflow storage (inches). This value represents the water in interflow storage at the beginning of the water year and should fall between zero and the Basic Interflow Volume Factor (BIVF). The parameter should be set at zero unless a rain has occurred within several days prior to October 1.

Note: If simulation is being run over a number of years, the starting moisture values for each year except the first are given in printout for the previous year as the ending moisture condition.

4.8 OPTIMIZATION

The process of parameter calibration, using a computer program such as OPSET (Liou, 1970), was described previously in Section 2.3. The program as developed and documented by Liou has been changed very little in previous uses by the IBM study team. In the study reported here it was not used at all; the Town Creek model had been calibrated previously. The following additional information is given to assist the reader who may wish to apply the model to another watershed.

4.8.1 PARAMETERS OPTIMIZED

The optimization process, a combination of automatic calibration and manual adjustments, as summarized previously, is used to quantify the following simulation model parameters.

1. BFRC, Base Flow Recession Constant, governs the rate at which groundwater flow recedes in the model.

2. IFRC, Interflow Recession Constant, governs interflow recession.

3. BUZC, Basic Upper Zone Capacity, is an index for estimating the capacity of the soil surface (upper zone) to store water in interception and depression storage.

4. SUZC, Seasonal Upper Zone capacity adjustment constant, is used to adjust upper zone variations in vegetation and cultivation.

5. LZC, Lower Zone Capacity, is an estimate of the capacity of the basin soil to hold water. Decreasing LZC in the model has the effect of increasing synthesized runoff.

6. BMIR, Basic Maximum Infiltration Rate, is the index used to control the basic rate of moisture infiltration. This is a parameter to which simulation accuracy is very sensitive, particularly as it affects storm peaks.

7. SIAC, the Seasonal Infiltration Adjustment Constant, is an evaporation-infiltration factor relating infiltration rates to evaporation rates to account for more rapid infiltration during warmer periods.

8. ETLF, Evapotranspiration Loss Factor, is an index used to estimate the maximum rate of evapotranspiration which could occur within the watershed under current conditions of soil moisture content.

9. BIVF, Basic Interflow Volume Factor, controls time distribution and quantities of moisture entering interflow. Increasing BIVF tends to reduce storm runoff peaks and extend hydrograph recession limbs.

10. NCTRI, Number of Current Time Routing Increments, is the number of subareas into which the basin should be divided, given 15 minutes or one hour separation between isochrones.

11. CSRX, Channel Storage Routing Index, is used to account for channel storage when channel flows are less than half of channel capacity (CHCAP). Channel storage effects are simulated by having the hydrograph time routed to the mouth of the watershed through an imaginary reservoir.

12. FSRX, Flood plain Storage Routine Index, is used to account for channel storage plus flood plain storage when streamflows are greater than twice the channel capacity. Between one-half and twice channel capacity, the program interpolate values between FSRX and CSRX.

13. CHCAP, Channel Capacity, is that value of streamflow, measured at the gage, at which a transition is made from channel routine to flood plain routine. In mountainous watersheds, this is not an oritical parameter, and OPSET seldom adjusts it.

After OPSET has adjusted the parameters listed above to achieve a "best match" based on mean daily streamflow, it is usually found that synthesized flood peaks fail to match observed peaks, in magnitude and/or time. Since the study attempted to address as wide a variety of applications as practicable, some manual "fine tuning" was undertaken to achieve an acceptable match between synthesized and observed flood peaks while maintaining an acceptable correlation between synthesized and observed mean daily and monthly flows.

This manual adjustment process requires some knowledge of the hydrologic processes occurring in the watershed and some subjective judgment. No firm rules or recipes have been developed, but the following are useful guidelines.

1. Overall results are affected by soil moisture capacities and infiltration rates (LZC, BUZC, BMIR) and their related seasonal adjustment constants (SUZC, SIAC).

2. Initial storages (LZS and UZS) can be varied to improve accuracy in the first two months of the multiyear simulation.

3. Summer storm peaks are affected more than winter storm peaks by changes in SIAC and SUZC, and the latter has more influence on mean daily flow in drier months than the former.

4. Consistent phasing errors (differences in times of occurrence between synthesized and observed flood peaks) can usually be reduced by adjustment if the number and sizes of subareas in the time-area histogram. Phasing errors which appear random are attributable to errors in input precipitation and/or evaporation data.

5. Since there are parameter interactions, all performance indices should be re-checked after any parameter adjustment and others re-adjusted as needed until a "best simulation" is realized.

The above comments apply equally to "fine tuning" the streamflow forecast model after a past run has been executed.

SECTION 5

CONTROL OPTIONS

The structure of the model permits the user to select a number of control options to meet his specific requirements. These include operating modes (long-term calibration, forecast and "fine tune"), inclusion or exclusion of snowmelt, selection of how evaporation is to be treated, and selection of input data. The sixteen control options, their meanings and their applications are listed in Table 5-1.

Table 5-1. CONTROL OPTIONS

OPTION NUMBER	APPLICATION	VALUE	EFFECT/MEANING
1	ENABLES OPERATOR TO REQUEST STORM DETAILS IN 15-MINUTE INTERVALS (SELDOM USED)	0 1	15-MINUTE DETAILS REQUESTED 15-MINUTE DETAILS NOT REQUESTED
2	DIVIDE EACH HOUR'S RAINFALL BY STORED STATISTICS AMONG 15-MINUTE PERIODS	0 1	DIVIDE RAINFALL EQUALLY DIVIDE RAINFALL STATISTICALLY
3	SELECT WAY IN WHICH EVAPORATION DATA ARE READ	0 1 2	READ DAILY EVAPORATION READ BY 10-DAY PERIODS USE ANNUAL EVAPORATION TOTAL, MNRD AND SUBROUTINE EVPDAY
4	SELECT DAILY FLOW ERROR TABLE	0 1	TABLE NOT REQUESTED TABLE IS REQUESTED
5	REQUEST 20 GREATEST RAINFALLS AND OVERLAND FLOWS	0 1	DATA NOT REQUESTED DATA REQUESTED
6	REQUEST DAILY VALUES OF SOIL MOISTURE STORAGE (TABLE OUTPUT)	0 1	DATA NOT REQUESTED DATA REQUESTED
7	INCLUDE SNOW IN ANALYSIS	0 1	SNOW NOT INCLUDED SNOW INCLUDED
8	SELECT PRECIPITATION PLOTS (ALWAYS SET=2 IN CURRENT MODEL)	0 1 2	NO PRECIPITATION PLOTS PRECIPITATION PLOTS ONLY PRECIPITATION PLOTS SUPERIMPOSED ON STREAMFLOW PLOTS
9	SELECT WHETHER DAILY RECORDED STREAMFLOWS ARE READ IN	0 1	RECORDED FLOWS NOT READ RECORDED FLOWS ARE READ
10	NOT USED IN PILOT FORECAST MODEL; ALWAYS SET=0	0	SINGLE-YEAR SIMULATION ONLY
11	INSTRUCT MODEL TO READ OR NOT READ DIVERSIONS (E.G., PUMPING IN OR OUT)	0 1	NO DIVERSIONS READ DIVERSIONS
12	SELECT STREAM ROUTING INTERVALS (DEPENDS ON BASIN SIZE)	0 1	15-MINUTE INTERVALS 1-HOUR INTERVALS
13	CHOOSE IF LENGTH OF TIME - AREA HISTOGRAM IS TO VARY WITH FLOW VELOCITY; ALWAYS SET=0	0 1	FIXED TIME-AREA HISTOGRAM (TAH) LENGTH VARIABLE TAH LENGTH
14	PRINTING RECORDED STREAMFLOW	0 1	DO NOT PRINT PRINT
15	SELECT MODE OF OPERATION FOR RIVER FORECAST APPLICATION	0 1 2	INITIALIZATION AND FORECAST RUN FORECAST RUN PAST ("FINE TUNE") RUN
16	SET INITIAL CONDITIONS IN DATA BANK FOR NEXT FORECAST RUN	0 1	DO NOT SET SET

SECTION 6

THE STREAMFLOW FORECAST MODEL

Early in 1972, IBM acquired a copy of the Kentucky Watershed Model through the courtesy of one of its developers, Dr. L. D. James, for use in investigations of remote sensing applications in hydrologic modeling. It has been modified several times since then by the addition of statistical analysis routines, plot outputs, multi-year simulations, and some efficiencies in inputting observed streamflow data. Additionally, it has been adapted to interface with terminal management software in two computer system installations. The principal modifications implemented for this study are those providing for a variable simulation period, a "past-run" capability, simulations based on three different precipitation inputs, and output formats suitable for CRT display. The forecast model subroutines are listed in Table 6-1.

6.1 THE MAIN PROGRAMS

The main program consists of the executive routine MAIN and the moisture accounting subroutines KWMAIN, WORSTC, ZEROP, FORCST, and PASTRN. The choice of moisture accounting subroutine depends upon the value assigned to CONOPT (15). If CONOPT (15) = 0, the KWMAIN subroutine is used to initialize the system, i.e., to read in all the parameter values, control options and initial values assigned to the variables. If CONOPT (15) = 1, the three subroutines WORSTC, ZEROP and FORCST are used to generate the three forecast runs and store the resulting simulated streamflow arrays. If CONOPT (15) = 2, the subroutine PASTRN is used to run a simulation (up to a selected day in the past) using actual precipitation inputs.

A flow diagram of the water accounting calculations is shown in Figure 6-1. The calculations are conducted in a series of nested daily, hourly, and 15-minute loops in the main program. The basic calculations are discussed as follows.

6.1.1 PRECIPITATION INPUTS

As discussed in Section 2.0, hourly rainfall values and auxiliary daily storage gage data are used to obtain a weighted hourly precipitation value for the watershed. This value, in turn, may be adjusted by the recording gage precipitation multiplier (RGPMB), depending on the degree of watershed calibration attained.

6.1.2 INTERCEPTION

All incoming precipitation is assigned to interception storage until the assigned interception storage parameter (VINTMR) is satisfied. Precipitation in excess of interception storage passes on to Upper Zone Storage (UZS) for the previous land surface area. Precipitation intercepted by the impervious land surface (FIMP) runs off the impervious surface to the channel. Interception storage is depleted by evaporation at the potential rate.

TABLE 6-1

STREAMFLOW FORECAST MODEL SUBROUTINES

NAME AND CONTROL OPTION*	FUNCTIONAL SUMMARY
MAIN	Executive subroutine; calls others in order, as governed by their respective control options.
KWMAIN CONOPT(15)=0	Moisture accounting subroutine, adapted from KWM; initializes model from beginning of the water year to the beginning of the forecast period.
WORSTC ZEROP FORCST CONOPT(15)=1	Moisture accounting subroutines for worst-case, zero and quantitative precipitation forecast, respectively; simulates streamflows through the forecast period.
PASTRN CONOPT(15)=2	Moisture accounting subroutine for actual precipitation input for up to 14 days in the past. Outputs actual and simulated streamflow for comparison and "fine tuning" of soil moisture storages.
OUTPUT	Formats Tables and Plots for past run and forecast run outputs for terminal CRT.
ATTEN	Model 2250 Terminal Operator Interface Routine.
BCDWD	Creates words for display/output from the decimal equivalents of given words; in BAL language.
CHANGE	Rearranges hourly precipitation inputs to normal order (October 1 - September 30), with February 29 taken into account. KWM normal order input is January 1 - September 30 (year 2), followed by October 1 - December 31 (year 1) and February 29.
CONVER	Converts hexadecimal data format to EBCDIC format for display on the 2250 terminal.
CORREL	Calculates correlation coefficients between daily and monthly simulated and observed streamflow.

*As applicable

TABLE 6-1

STREAMFLOW FORECAST MODEL SUBROUTINES (CONTINUED)

NAME AND CONTROL	FUNCTIONAL SUMMARY
DAYNXT	Calculates the number in the water year of the next day, taking leap years into account.
DAYOUT	Formats and prints tables of daily values.
DICTRY	Formats dictionary (choice of displays) for display.
DLPO	Residue routine - not used.
DMSG	Residue routine - not used.
DSMRG	Formats the updated test line displayed on the 2250 terminal.
DSREAD	Performs the contiguous read and display of the Hydrology Data Base on the 2250 terminal.
DSUPDT	Performs the actual update of Hydrology Data Base which is displayed on 2250 terminal.
DSWRITE	Performs the contiguous write to disk of the updated Hydrology Data Base.
DVAR	Display/Option Processing Module; allows the calling program to communicate with the user(s); processes light pen and keyboard entries.
EVPDAY CONOPT(3)=2	Determines pan evaporation totals for each day of the water year from mean annual evapotranspiration total.
GSPD	Common Data Block module from which Hydrology Simulator updates and accesses stored data.
INT	Finds the integral of a given hydrograph (i.e., runoff) and plots the hydrograph. Print plots and tabular plots are available.

TABLE 6-1

STREAMFLOW FORECAST MODEL SUBROUTINES (CONTINUED)

NAME AND CONTROL OPTION	FUNCTIONAL SUMMARY
INTEG	Performs the actual integration for the hydrograph plot represented on display and printer.
TABLE	Performs a table look-up and linear interpolation.
LABELS	Creates the labels associated with the coordinates of hydrograph plots.
PL360	Plots N points using the coordinates from two input arrays. Scalers are adjusted for maximum and minimum values in both arrays. This output is represented on printer plots only.
PL4020	Performs the generation of hydrograph plots to be displayed as specified by the parameter argument list.
PLOTID	Performs the necessary 2250 terminal I/O and Data Set initialization.
PLOTEND	Terminates on-line status of the 2250 terminal.
POINTS	Performs the actual plotting of points for requested hydrograph.
PR360	Performs the printing of requested hydrograph plot.
PRECHK	Checks precipitation - streamflow anomalies and adjusts precipitation where necessary.
PREPRD CONOPT(3)=1	Divides hourly precipitation among 15 minute accounting periods according to an algorithm developed by Liou (1970).
RTVARY	Increase flow velocity (in routing) as discharge increased.
SGRID	Creates the coordinates necessary for requested hydrograph.

TABLE 6-1

STREAMFLOW FORECAST MODEL SUBROUTINES (CONTINUED)

NAME AND CONTROL OPTION	FUNCTIONAL SUMMARY
SHIFT1	Performs the actual creation and storing of Display Dictionary elements.
SNOMEL CONOPT(7)=1	Moisture accounting in the snowpack. Calculates equivalent precipitation input from snowmelt. Obtained from W. D. Striffler (1973).
STAT	Statistical analysis of simulated vs recorded streamflow. Calculations include sums, maxima, means, sums of squares, variances and correlation coefficients.

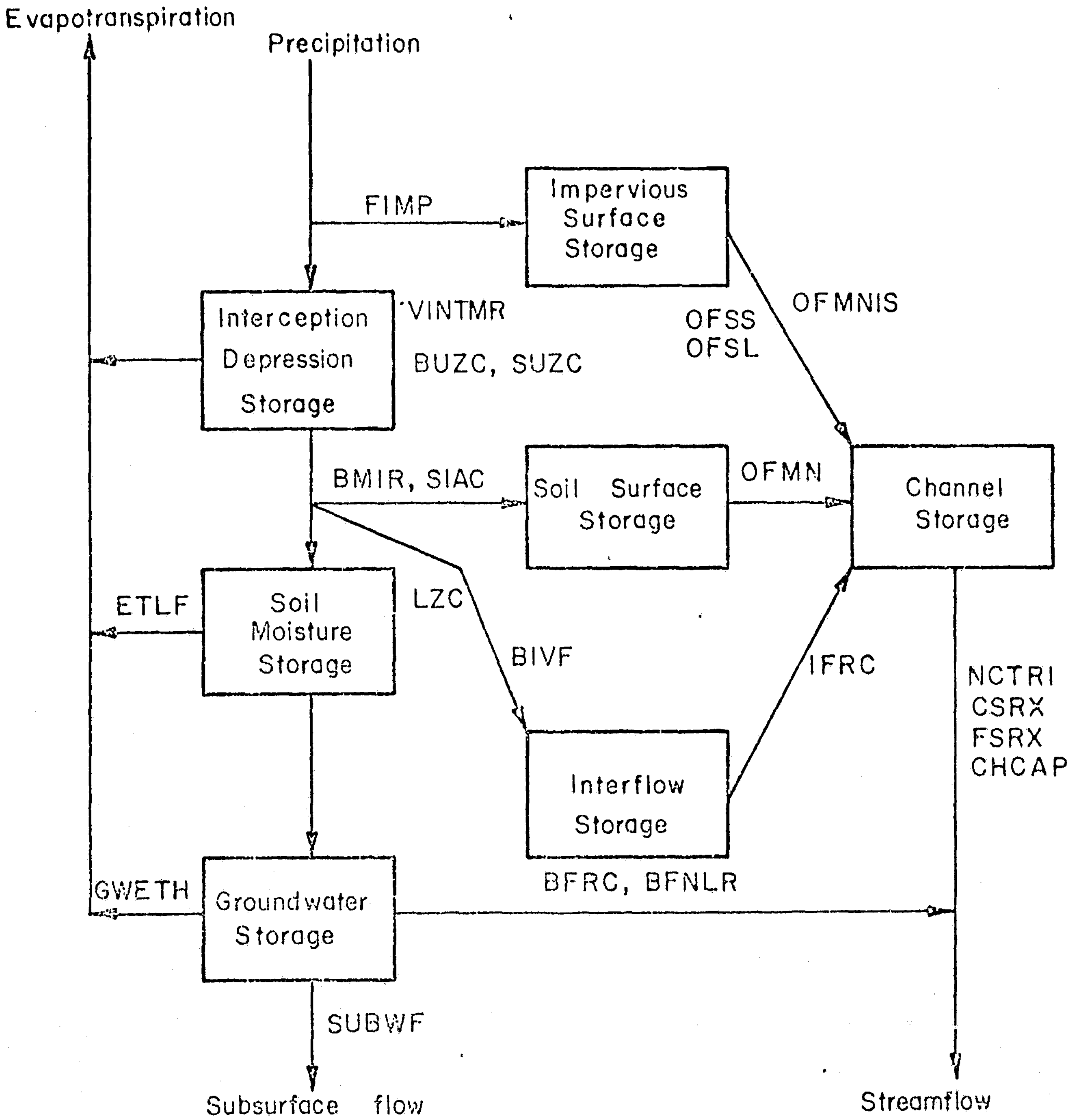


Figure 6-1. Water Accounting in the Stanford Watershed Model (Ross, 1970)

6.1.3 UPPER ZONE STORAGE

Precipitation in excess of interception storage reaches the ground surface where it either infiltrates into the soil or, if precipitation rates are greater than infiltration rates, accumulates on the surface as depression storage or overland flow detention storage. Water infiltrating will go either to lower zone storage or interflow storage. The distribution of surface water into the various components (surface detention, interflow storage, infiltration) depends upon the upper zone storage capacity (UZC), the lower zone storage ratio (LZS/LZC), the basic interflow volume factor (BIVF), and the seasonal adjustment factor (SIAC). The model for distributing surface water is illustrated in Figure 6-2.

The upper zone storage capacity is calculated as:

$$UZC = SUZC\ (AEX90) + BUZC(e^{-2.7\ LZSR})$$

where SUZC is the seasonal upper zone storage

AEX90 is an evaporation index.

BUZC is the basic upper zone storage capacity.

and LZSR is the ratio of current lower zone storage to the lower zone storage capacity (LZS/LZC).

The fraction of moisture retained (FMR) is then calculated as a function of the Upper Zone Storage Ratio (UZS/UZC) accoring to the relationship shown in Figure 6-3. Upper zone storage is depleted by both infiltration (delayed) and evaporation. Delayed infiltration is calculated as:

UZINLZ = 0.003 BMIR * UZC * UZINFX ** 3.0

where UZINLZ = current upper zone infiltration to lower zone.

BMIR = basic maximum infiltration rate.

UZC = upper zone storage capacity.

and UZINFX = upper zone infiltration index.

= (UZS/UZC) - (LZS/LZC).

Upper zone losses to evaporation are at potential rate as long as storage is present.

6.1.4 ACTIVE INFILTRATION

Water in excess of the upper zone storage capacity will be distributed to interflow storage, infiltration, or overland flow according to the model in Figure 6.2. The proportion of water going to lower zone and ground water storage depends upon the infiltration rate. As illustrated in Figure 6.2, infiltration rates are distributed linearly over the watershed from 0 to a maximum (BMIR) at some point on the watershed. The basic maximum infiltration rate (CMIR) for each 15-minute period during a storm:

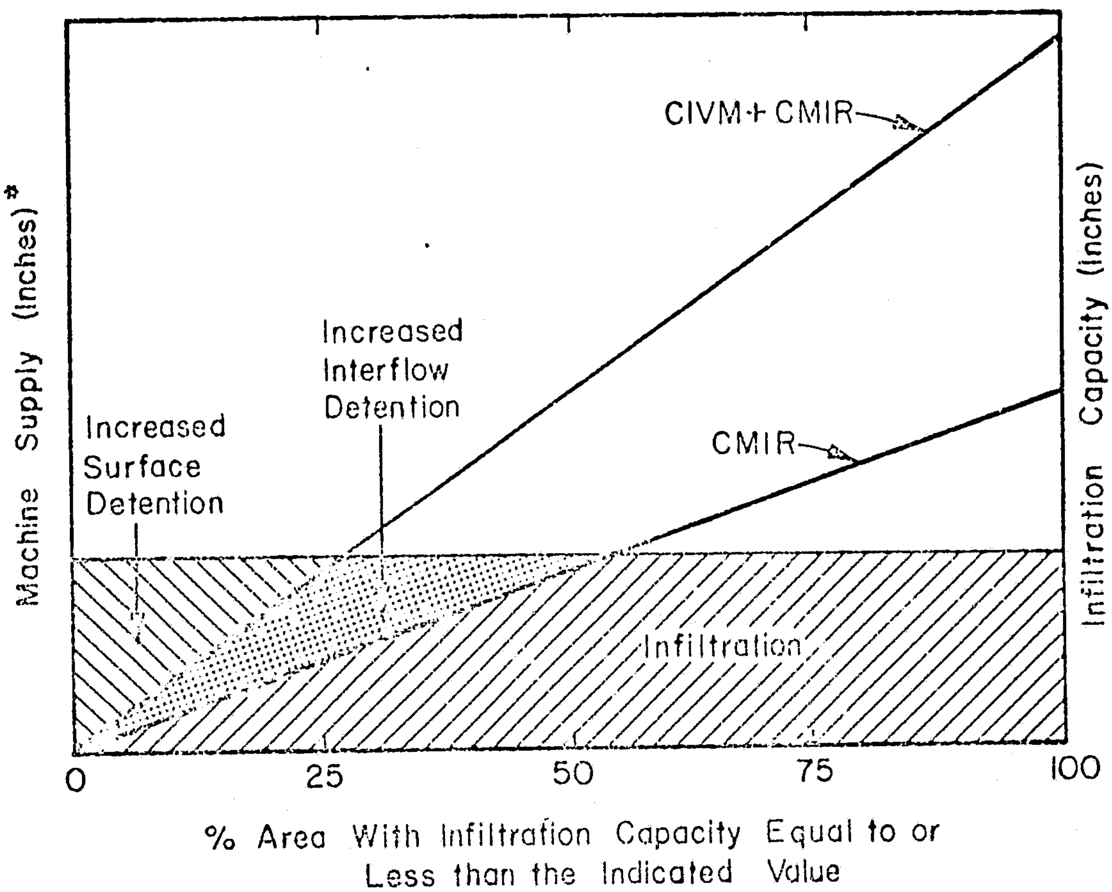


Figure 6-2. Model for Estimating Infiltration Capacity (Ross, 1970)

* Rainfall Passing Through the Upper Zone Plus Holdover Direct Runoff.

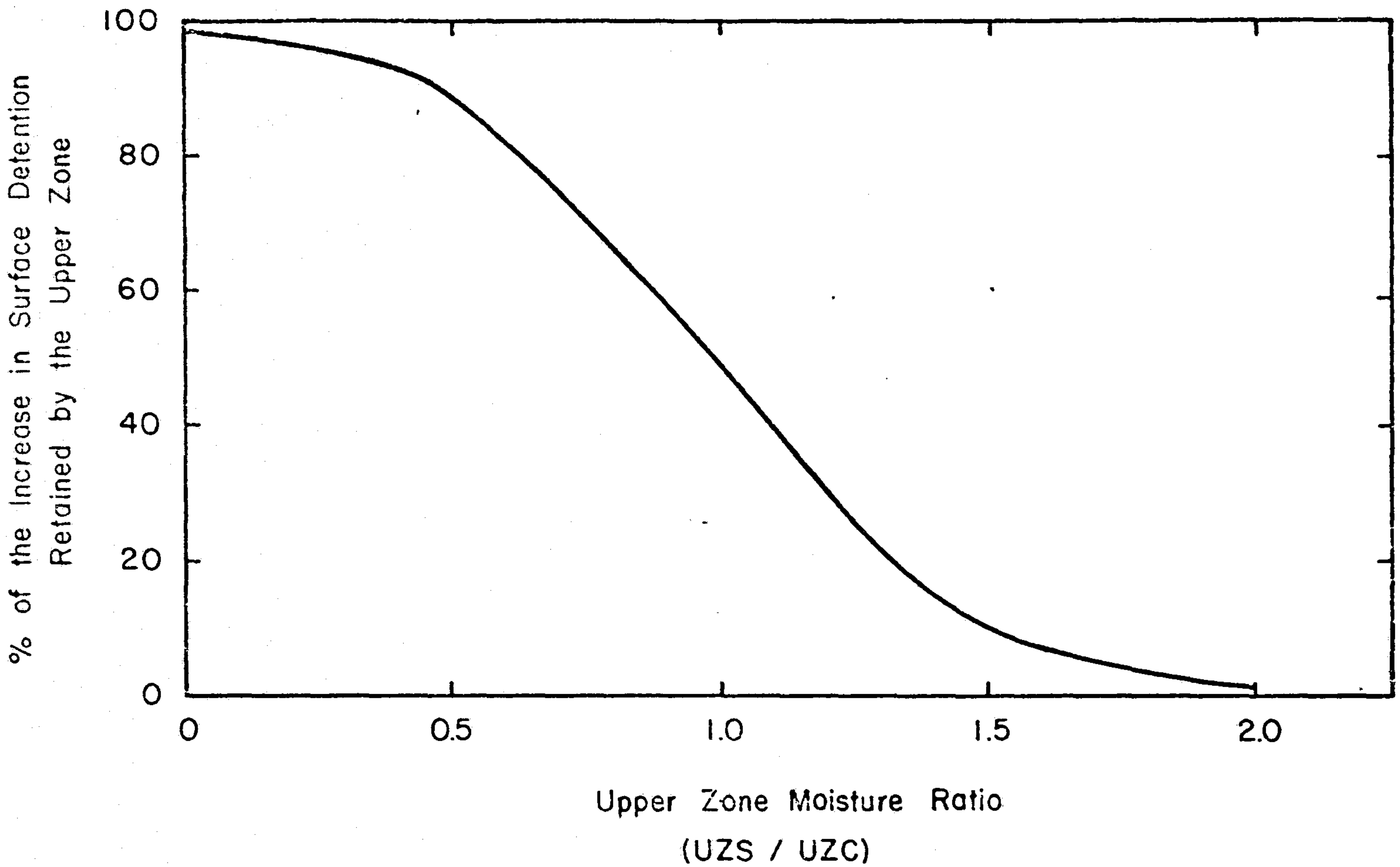


Figure 6-3. Model for Estimating the Upper Zone Storage Component of Surface Detention (Ross, 1970)

CMIR = 0.25 * SIAM * BMIR/(2.0 ** EID)

where CMIR = adjusted current maximum infiltration rate.

SIAM = seasonal infiltration adjustment multiplier.

= 1.2 ** SIAC

and EID = exponent of infiltration rate decay with increased soil moisture content

= 4.0 * LZSR for LZSR values less than 1.0

The proportion of water going to interflow storage is also calculated as part of the active infiltration calculation:

CMIV = BIVM * 2.0 ** LZSR

where BIVM = basic interflow volume parameter.

LZSR = lower zone storage ratio (LZS/LZC).

6.1.5 LOWER ZONE AND GROUND WATER RECHARGE

The portion of the water infiltrated into the ground will either go into lower zone storage (LZS) or pass through to ground water storage (GWS) depending on the lower zone storage ratio (LZS/LZC). The amount of water infiltration to the lower zone storage is calculated as:

FMR = (1.0/(1.0 + LZRX)) ** LZRX

where FMR = fraction of moisture retention in lower zone

and LZRX = lower zone moisture retention index.

= 1.5 * ABS(LZSR - 1.0) + 1.0.

The proportion going to ground water is calculated as:

PGW = (1.0 - FMR) * UZINLZ * (1.0 - SUBWF) * FPER

where PGW - percolation to ground water.

FMR = fraction of moisture retention

UZINLZ = current upper zone infiltration to lower zone.

SUBWF = subsurface water flow out of basin.

and FPER = fraction of watershed being pervious.

The recharge to ground water as a function of the lower zone storage ratio is shown in Figure 6-4.

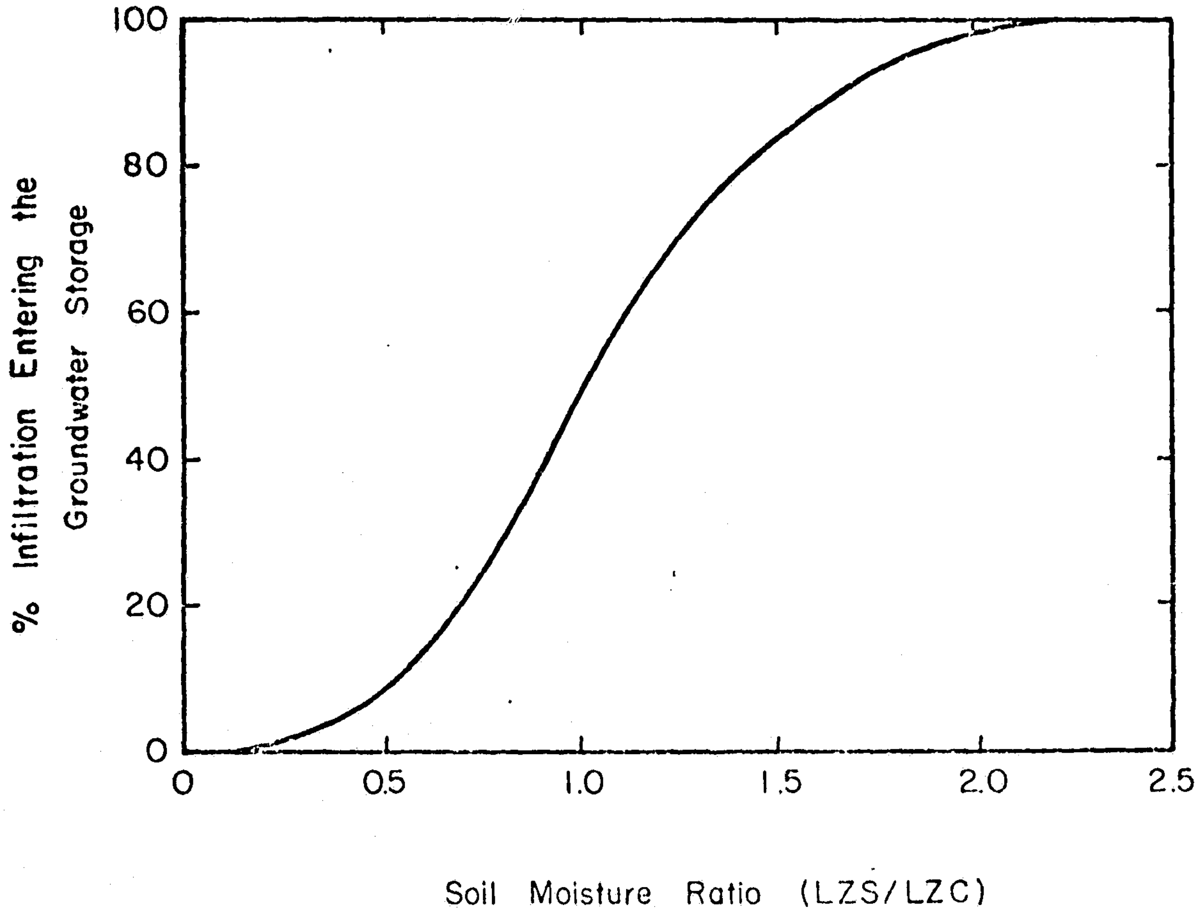


Figure 6-4. Model for Estimating Infiltration Entering Groundwater Storage (Ross, 1970)

6.1.6 OVERLAND FLOW

Overland flow calculations are based on a modified version of the Chezy-Manning Equation:

$$q = \frac{1.486}{n} \quad y^{5/3} s^{1/2}$$

where q = flow rate in ft^3/sec/ft.

y = depth in feet at the lower edge of the flow plane.

s = slope of the flow plane in ft/ft.

and n = Manning's roughness coefficient.

Crawford and Linsley (1966) used this equation to obtain a relationship between surface detention at equilibrium, the supply rate to overland flow, the length and slope of the flow plane, and Manning's n. In the Kentucky model the overland flow calculation takes the form:

```
OFR = 0.25 * OFRF * ((OFUS + PEAI) * 0.05) ** 1.67 * ((1.0 + 0.6)
      * ((OFUS + PEAI) / (2.0 * EQD)) ** 3.0) ** 1.67
```

where OFR = current overland flow runoff (inches/hour).

OFRF = overland flow routing factor.

= 1020.0 * (OFSS ** 0.5)/(OFMIN * OFSC).

(OFUS + PEAI) / 2.0 = detention volume (corresponding to y).

and EQD = equilibrium depth of overland flow

= EQDF * ((PEAI - OFUS) ** 0.6)

A similar calculation is used for overland flow from both previous and impervious surface within the watershed.

6.1.7 INTERFLOW

The volume of water entering interflow storage was calculated previously. Interflow storage is depleted according to a logarithmic decay function:

SPIF = IFRL * IFS

where SPIF = synthesized period interflow.

IFRL = log if interflow period recession constant.

and IFS = interflow storage.

6.1.8 GROUND WATER FLOW

The volume of water entering ground water storage was calculated previously. The flow from ground water storage to the stream is calculated as :

CBF = GWS * BFRL * (1.0 + BFNRL * BFNX)

where CBF = current base flow.

GWS = ground water storage

BFRL = the log of the base flow recession constant.

BFNRL = the log of the nonlinear base flow adjustment factor

and BFNX = current value of the nonlinear base flow adjustment factor.

Losses from ground water storage may occur by deep seepage or flow out of the basin, and by direct evaportranspiration losses. Ground water flow from the basin is specified in the input parameter SUBWF and adjustments made to ground water storage (GWS). ET losses from ground water are determined from the input parameter GWETF which essentially designates the proportion of the watershed from which ground water is evaporating and calculate ET at potential rates from that area.

6.1.9 EVAPOTRANSPIRATION

Evapotranspiration losses from the watershed occur at various locations and at various rates. Evaporation from interception storage, upper zone storage, ground water storage (when indicated) and open water surfaces is calculated at the potential rate. Evapotranspiration losses from the remainder of the basin are calculated at a variable rate depending on the availability of soil water. Potential evapotranspiration is assumed equal to lake evaporation which is determined from pan evaporation data (DPET) and a monthly pan coefficient (EPCM). As discussed in Section 3, pan evaporation data can be input as daily values, average values over a 10-day periods, or a mean annual ET value (EPAET). If the latter is used, subroutine EVPDAY is called and the mean annual value distributed over the days of the year.

Two adjustments are made to the potential evapotranspiration rate (PET). First, if the daily maximum temperature (DMXT) minus 4.0 times the elevation difference (ELDIF) is less than 40.0, PET is set at zero. Second, if the basin is snow covered, PET is only calculated from the portion of the basin covered by forest (PET = FFOR * PET).

In the model, PET is first taken from Interception Storage, Ground Water Storage (if specified), and upper zone storage at the potential rate. If PET remains, it is taken from lower zone storage according to the form:

SET = PET * (1.0 - PET/(2.0*ETLF * LZSR))

where SET = current hourly soil evaporation

PET = current daily potential evaporation.

ETLF = the ET loss factor (parameter).

and LZSR = lower zone storage ratio.

The term (ETLF * LZSR) defines the maximum rate at which evaporation can occur, and determines the distribution of evaporation from the watershed according to the model illustrated by Figure 6-5.

Other evaporation losses from the watershed include Daily Potential Snow Evaporation (DPSE) which is calculated separately from watershed PET and Evaporation from open water surfaces (FWTR).

Snow evaporation is calculated as a function of the daily minimum temperature and the daily potential evaporation data array (DPSE). If the daily minimum temperature is less than 32°F., snow evaporation is equal to DPSE. Evaporation from the free water surface on the watershed is calculated as:

$$HSE = (FWTR * PET)/12.0$$

where HSE = hourly stream evaporation

FWTR = proportion of watershed covered by water

PET = potential evaporation

HSE is calculated between the hours 9:00 a.m. and 9:00 p.m.

6.1.10 ROUTING

The overland flow and interflow components are delivered to the channel system and must be routed through the channel system to simulate the hydrograph at the basin outlet. Linsley and Crawford (1966) adapted an empirical method of Clark (1945) using the time-area histogram discussed in Section 3.1. The basic premise of the method is that the time-area histogram represents the time distribution of outflow from the watershed. Inflow into each element of the time-area histogram is routed through the element using the relationship:

$$O_2 = \bar{I} - CSRX\ (\bar{I} - O_1)$$

where O_2 is the outflow at the end of the time interval

O_1 is the outflow at the begining of the time interval

$\bar{I}$ is the average inflow into the reservoir during the time interval

and CSRX is the channel storage routing index.

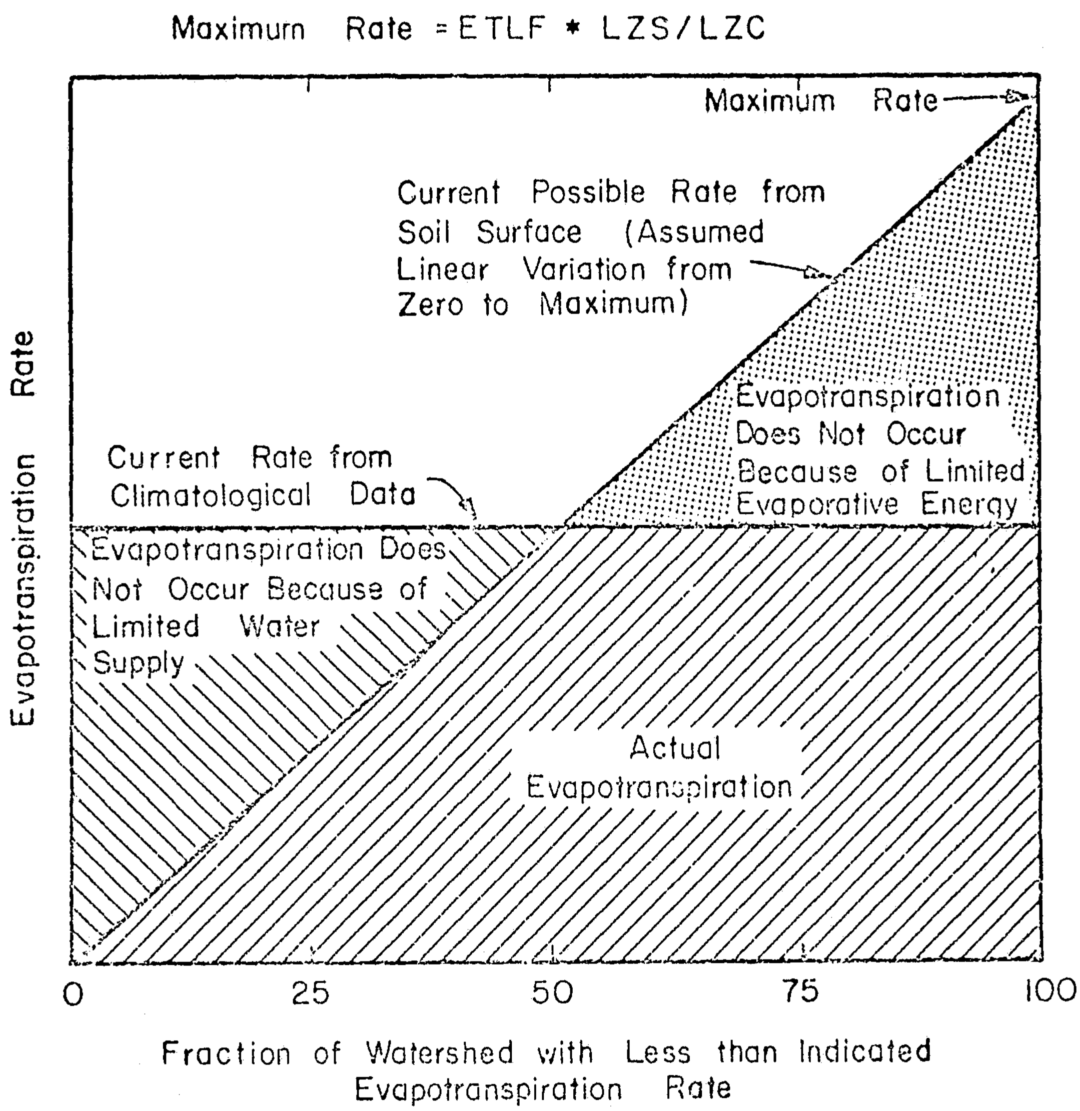


Figure 6-5. Model for Estimating Actual Evapotranspiration (Ross, 1970)

The program provides for three chnannel flow situations: (1) if the synthesized channel flow is less than half the channel capacity (CHCAP), the channel storage routing index (CSRX) is used in the above calculation. (2) if the synthesized flow is over twice the channel capacity, indicating that the flow is out of the banks, the flood plain storage routing index (FSRX) is used; and (3) if channel flow is between 0.5 and 2.0 CHCAP, a value is interpolated between CSRX and FSRX.

The routing calculations discussed assume that routing through the channel system is linear. If there is reason to believe that the time routing of storm hydrographs is nonlinear, a provision for increasing downstream velocity is provided in subroutine RTVARY.

6.2 SUBROUTINE SNOMEL

The snow melt subroutine is a simplified version of the Stanford Snow melt Model of Anderson and Crawford (1964) devised by Colorado State University. The model calculates snow melt as a function of air temperature with adjustments for seasonal radiation incidence and albedo. The model also maintains a continuous accounting of physical snow pack characteristics which enter into the snow melt accounting. The snow melt model is shown in Figure 6-6. It was not used in modeling the Town Creek basin, where snow is seldom of measurable effect.

6.2.1 ELEVATION EFFECTS

Precipitation - Precipitation is assumed uniform over the watershed. This is calculated by determing the ratio of gage catch to watershed precipitation. This is an erroneous assumption for high mountain watersheds, especially large watersheds where precipitation varies strongly with elevation. No elevation correction is applied to the model for precipitation.

Temperature - Hourly temperature values are used in the calculations to determine whether precipitation is rain or snow, and also to calculate snow melt. Hourly temperature values are estimated according to a sine wave distribution between two representative extremes, the maximum and minimum. It is assumed that the maximum and minimum air temperature occur at 4:00 p.m. and 4:00 a.m. respectively. The recorded maximum and minimum are adjusted for elevation to give maximum and minimum values for the basin. Adjustments are based on the following relationships:

1. At 4:00 p.m. the lapse rate = 4.00/1000 feet.

2. At 4:00 a.m. the lapse rate = 0.00/1000 feet.

3. During precipitation events, the lapse rate = 3.30/1000 feet.

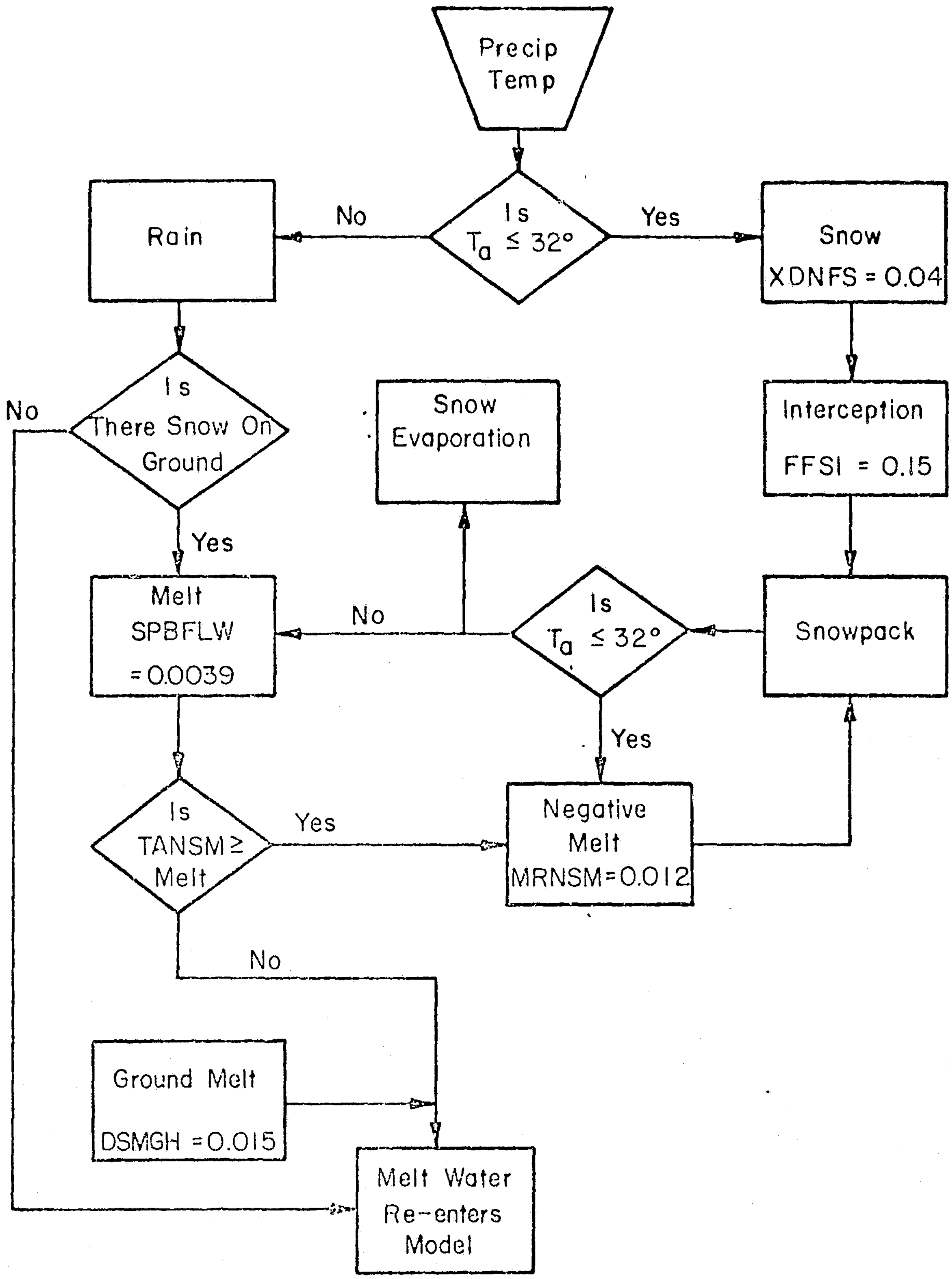


Figure 6-6. Snowmelt Calculations in the Kentucky Watershed Model

6.2.2 SNOW ACCUMULATION

Accumulation - Temperature is not a perfect index to the form of precipitation but is assumed best. The subroutine uses a single temperature index to determine the form of precipitation. If the hourly temperature (TEH) is above 32°F, precipitation occurring during this hour is rain, while if TEH is 32°F or less, precipitation is snow. If experimental data or local experience indicate a different index value would be more appropriate, the index value can be changed in the subroutine.

Gage Catch Corrections - Precipitation gages are not as efficient in catching snow as they are in catching rain. If a consistent low runoff volume is synthesized, a gage catch correction may be added by setting the value of the snow precipitation multiplier (SPM) greater than 1.0.

6.2.3 LOSSES FROM THE SNOW PACK

Interception Losses - Snow caught and held by the forest canopy is assumed proportional to the proportion of forest cover. The calculations are:

$$FSIL = FFSI * FFOR * HSF$$

Where FSIL = Hourly snow interceptive loss

FFSI = Fraction intercepted by the forest

FFOR = Fraction of forest cover in the watershed.

HSF = Hourly snowfall.

This loss is deducted from the hourly snowfall before it reaches the ground.

Evaporation Losses - Evaporation losses from the snow pack are calculated separately from other evaporation losses and deducted from the snow pack water content (SPTW). Snow evaporation is determined using a curve of average snow evaporation over the year (DPSE data array). Snow evaporation is equal to the daily value if the daily minimum temperature is less then 32°F.

6.2.4 SNOW PACK CHARACTERISTICS

The snowmelt subroutine also keeps a running account of the snow pack characteristics since melt rates depend upon the state, or "ripeness" of the snow pack. These characteristics include:

Snow Albedo - The snow albedo correction is considered to vary with time and will range from a maximum value (fresh new snow) to a minimum value (ripe, melting snow) over a 15-day period. This array of values is included as a data array (FIRR). Under melt conditions, the snow albedo index (SAX) will increase by one each day until a maximum SAX of 15 is reached. Similarly, if a fresh snow falls on the snow pack, the albedo under is decreased by one

each day that fresh snow falls until a SAX value of 0.0 is reached. Rain on snow will also change the albedo index, thus if the hourly rainfall is greater than the precipitation index for changing snow albedo (PSCSA), the albedo index will increase by 1.0. The albedo value (FIRR) is used to adjust the melt rate.

<u>Density of New Snow</u> - The density of new snow will vary according to the air temperature. Snow falling in a cold, dry atmosphere will be light and fluffy, while snow falling in a warmer environment will be wet and heavy. The density of new snow is determined by the index density of new fallen snow (XDNFS). If the hourly air temperature (TEH) is less than 0°F, the density of new snow is set at the index density. If the hourly air temperature is greater than 0°F:

$$\mathrm{DNFS} = \mathrm{XDNF} + (0.1(\mathrm{TEH}))**2.0$$

where DNFS = Density of New Snow

XDNFS = Index Density of New Snow

TEH = Hourly Air Temperature.

<u>Compaction and Setting of the Snow Pack</u> - The snow melt subroutine assumes that all changes in the snow pack occur when new snow is added. When new snow is added to the snow pack, the added weight causes a certain amount of setting and compaction. This is calculated as:

$$\mathrm{SDEPTH} = \mathrm{SDEPTH} - ((\mathrm{PRH}* \mathrm{SDEPTH}/\mathrm{SPTW}) * (0.10 * \mathrm{SDEPTH}) ** 0.25))$$

where SDEPTH = Snow Depth

PRH = Precipitation Recorded Hourly

SPTW = Snow Pack Total Water Content.

Reduction in snow depth also occurs due to snow melt. This is calculated as a function of hourly snow melt (HSM), the density of frozen water in the snow pack (SFMO), and the snow albedo index (SAX). Thus, for maximum melt conditions (SAX = 15.0):

$$\mathrm{SDEPTH} = \mathrm{SDEPTH} - (\mathrm{HSM}/\mathrm{SFMD})$$

when SDEPTH = Snow Depth

HSM = Hourly Snow Melt

SFMD = Snow Frozen Moisture Density.

For snow melt conditions less than maximum, (SAX 15.0) the SFMD factor is reduced proportionately.

Snow pack densities are also recalculated as a result of compaction and melt processes, as a function of snow depth and water content.

STMD = (SPTW + SPLW)/SDEPTH

and SFMD = SPTW/SDEPTH

where STMD = Snow Pack Total Moisture Density

SFMD = Snow Pack Frozen Moisture Density

SPTW = Snow Pack Total Water Content

SPLW = Snow Pack Liquid Water Content

SDEPTH = Snow Depth

The snow pack liquid water holding capacity is determined as:

SPLWC = SPBFLW * SPTW

where SPLWC = Snow Pack Liquid Water Holding Capacity

SPBFLW = Snow Pack Basic Maximum Fraction in Liquid Water

SPTW = Snow Pack Total Water Content.

Snow melt first goes to satisfy the liquid water holding capacity before being released to the soil.

6.2.5 SNOW MELT

Snow melt occurs from three basic heat sources: The heat in the air over the watershed, heat in rain falling on the snow pack, and heat stored in the ground beneath the snow pack. Snow melt is calculated separately from these sources.

Hourly Snowmelt - Hourly snow melt is calculated as a function of the basic degree hour melt factor (BDDFSM), and is adjusted for snow pack albedo and seasonal radiation incidence. The basic equation is:

HSM = (TEH - 32°) * BDDFSM * RICD * (1-(1-FFOR) * FIRR + (TEH - 32°))*

(PRH/144) * SPTW/SPTWCC

where HSM = Hourly Snow Melt

TEH = Hourly Air Temperature

BDDFSM = Basic Degree Hour Melt Factor

RICD = Radiation Incidence Correction Factor

RICY* 1 + (FDAY/10.0)

FFOR = Fraction of Forest Cover on the Basin

FIRR = Fraction of Radiation Reflected

PRH = Precipitation Recorded Hourly

SPTW = Snow Pack Total Water Content

SPTWCC = Total Snow Pack Water Content to Cover the Basin.

In the Kentucky version of the snowmelt model, the model assumes that the base temperature for snow melt is 32°. Thus, if the hourly air temperature is greater than 32°, snow melt will occur. The base temperature of 32° was selected for convenience and may not be the best point for some watersheds. If experience or research suggests a more appropriate base, it can be easily changed in the program.

Since the snow melt factor BDDFSM represents the maximum melt rate, several adjustments are required. The first is to adjust for the incidence of radiation over the year. Since snow melt is not simply the result of sensible heat exchange from the air mass over the watershed, but also due to short wave radiation, the amount of radiation incident upon the snow pack is important. This adjustment is applied in the RICY array which essentially represents the distribution of solar energy over the year. Not all short wave radiation is absorbed, but much is reflected depending on the albedo of the snow pack. The second adjustment is for the albedo of the snow pack and the proportion of forest cover on the watershed. The albedo reduction factors are provided in the data array FIRR discussed in Section 3.2.

Snow melt due to rain falling on this snow pack is also determined. The program assumes that rainfall has the same temperature as the air above the snow pack. While this may not be true, it is a reasonable assumption for the snow melt calculation. Again, the base temperature is set at 32° and the melt resulting from rainfall is determined.

The final adjustment to hourly snow melt on the basin is for partial snow cover conditions. The program assumes that if the total water content of the snow pack is greater than some index value (SPTWCC), melt is occurring uniformly over the basin. However, if the snow pack water content falls below the index value, snow cover is assumed not complete over the basin, and melt is reduced proportionately.

<u>Negative Snow Melt</u> - Since snow melt only occurs when air temperatures are 32°F or greater, the program assumes that negative melt or freezing occurs in the snow pack when temperatures are below 32°. In reality, the re-freezing of liquid water and chilling of the snow pack are highly complex processes depending on the density of the snow pack, the water holding capacity, the liquid water content, the structure, and heat content of the snow pack. In the model the re-freezing of liquid water and chilling of the snow pack are considered together as negative snow melt. Because of the complexities of the processes, an empirical function relating negative snow melt to air temperatures is used (Anderson and Crawford, 1964).

The routine assumes that negative snow melt occurs at some maximum rate (MRNSM) for the first twelve hours. Thereafter, the negative snow melt is calculated as:

NSM = ((5.0 * MRNSM) ** (1.3 + 2.0 * TANSM))

where NSM = Negative Snow Melt

MRNSM = Maximum Rate of Negative Snow Melt

TANSM - Total Accumulated Negative Snow Melt

Total accumulated negative snow melt cannot exceed .08*SPTW.

Snowmelt From Ground Heat - In some watersheds, snow melt due to heat in the ground is a significant source of melt. The Stanford Snow Melt Model assumes that melt from ground heat is constant (DSMGH). The total snow water content is reduced by this amount each day and added to overland flow.

6.3 OTHER SUBROUTINES

Most of the subroutines not previously discussed in this section were adequately summarized as to function in Table 6-1. Some of them are given additional attention in the following paragraphs.

6.3.1 SUBROUTINE READ

Subroutine READ is a machine language subroutine for reading data into the program. Since large amounts of data are required in the program, READ permits the use of unformatted data, data cards with labels or comments on the data card, and the insertion of comment or blank cards in the data deck.

Basically, a CALL READ statement in the program deck should include the name of the variable (or variables) being read and the number of values to be read. For example, in CALL READ (CONOPT (1, 15)), the computer would read the next 15 values in the data deck whether they are on one card or 15 cards and store the values in array CONOPT (15). Similarly, a CALL READ (GWS, UZS, LZS, BFNX, IFS) statement would cause the next five values to read and store with the appropriate label.

A few simple rules should be observed in the use of the subroutine.

1. Use only on numeric data, either real or integr. READ cannot read alphanumeric data. Thus, title cards, etc. must be formatted as usual. In addition, alphanumeric cards must follow a numerical data card and cannot follow a comment card.

2. Data values must be punched on the cards and placed in the deck in the order in which they are called.

3. The type of variable value should agree with the designation of the variable names, i.e., real, integr.

4. A space should be left between consecutive values although no other spacing requirements are required.

6.3.2 SUBROUTINE DAYNXT

Subroutine DAYNXT is a simple routine which calculates the next day of the year. Since most of the daily moisture accounting is calculated and stored according to the number of the day, DAYNXT determines the number of the next day of the year making appropriate adjustments for Leap Years and beginning a new year.

6.3.3 SUBROUTINE DAYOUT

Subroutine DAYOUT is a subroutine called in printing out the tables of daily synthesized streamflow values, daily recorded streamflow values, and daily soil water values. The subroutine converts daily values from calendar year day values (VDCY) to month day values (VDMD) and prints the daily values arranged by month in the water year month order.

6.3.4 SUBROUTINE EVPDAY

Subroutine EVPDAY determines the daily potential evaporation (DPET) for each day of the year, given the mean annual potential evapotranspiration (EPAET). The subroutine is called when CONOPT 3 = 2 and requires EPAET and MNRD data cards. Liou (1970) suggests that the subroutine is most valuable when a number of watersheds from the same region are to be tested and evaporation data are scarce. The tests can be run without the necessity of developing detailed evaporation data for each watershed.

Basically, the subroutine consists of an array of fractions for each day of the year which determines the potential evaporation for that day. When CONOPT 3 = 2, the main program first calculates the annual potential evaporation which would occur if no rain were recorded:

$$EMAET = EPAET \left(\frac{365 + MNRD}{404}\right)$$

where EPAET and MNRD are input data values for the estimated potential annual evaporation and the mean number of rainy days. Liou (1970) indicates that the denominator would be (365 + 0.5 * MNRD) if there were an equal likelihood of rain occurring during any day of the year, but that 404 works better in Kentucky where more rain occurs during the winter. The EMAET value is then multiplied by each of the daily evaporation fractions to obtain daily evaporation values. Ross (1970) discusses a method of determining the evaporation fraction used in EVPDAY and suggests that similar procedures should be used to obtain values for areas remote from Kentucky. Liou (1970) indicates that the Kentucky array seemed to work well in California but should be slightly more peaked for more northern latitudes and higher elevations.

6.3.5 SUBROUTINE PREPRD

The Kentucky Watershed Model requires hourly rainfall data to simulate infiltration and surface runoff processes. Although hourly data is read into the program, the moisture accounting subroutine divides the hourly value into four equal 15-minute values for the infiltration and overland flow calculation. This is strictly a convenience for a simplified calculation since rainfall is

seldom uniform for an hour period. On large basins with runoff lag times of several hours, this simplification is satisfactory. However, on small watersheds or in situations where the hourly distribution of rainfall might influence the timing and magnitude of the outflow hydrograph, it is desirable to have precipitation data for periods less than an hour, or, to distribute hourly precipitation according to a probable distribution. Subroutine PREPRD provides the user with the option (CONOPT 2 = 1) of subdividing hourly rainfall into non-uniform 15-minute quantities according to an average distribution.

The average distribution was determined from a curve showing the average distribution of 6-hour precipitation developed by the Bureau of Reclamation (1960) (Liou, 1970). The distribution of hourly rainfall determined by Liou (1970) is into fractions of 0.46, 0.28, 0.16, and 0.10. The proper sequencing of these fractions during the hour is a problem. PREPRD distributes rainfall according to the following conditions:

1. If the current hour rainfall is greater than both the previous and the following hour, the distribution is 0.10, 0.28, 0.46, and 0.16 of the hourly total.

2. If current hourly rainfall is less than both the previous and the following hour, the distribution is 0.28. 0.10, 0.16 and 0.46.

3. If the current hour rainfall is greater than the preceding hour but less than the following, the order is 0.10, 0.28, 0.16, and 0.48.

4. If current hour rainfall is less than the preceding but greater than the following hour, the order is 0.46, 0.16, 0.28, and 0.10.

6.3.6 SUBROUTINE RTVARY

In the channel routing procedure, the program assumes that the flow velocity is relatively uniform throughout the channel length and for all flow conditions. Since this may not be a valid assumption in many watersheds, especially during stormflows, subroutine RTVARY provides the user with a routine to increase flow velocities as discharge increases. The equation is:

$$V = K*Q^{(EXQPV)}$$

where V = flow velocity

K = constant

Q = channel flow volume

and EXQPV = an exponent of flow proportional to velocity.

The primary effect of this subroutine is to magnify flood peaks.

SECTION 7

OUTPUTS

Examples of streamflow forecast model outputs appear in this section.

- o Tables of hourly streamflow values in cfs for worst case, zero precipitation and quantitative forecasts.
- o Three superimposed hydrographs, one for each forecast case
- o Table of daily precipitation, surface runoff, interflow, base flow and storm runoff for each forecast case.
- o Table of peak flow, time of peak, total runoff and total precipitation for the forecast period
- o Soil moisture storages and parameters at the end of each day for each forecast case. There are interflow (IFS), upper zone (UZS), lower zone (SZS), ground water (GWS), baseflow nonlinear recession index (BFNX), seasonal infiltration adjustment multiplier (SIAM) and upper zone capacity (UZC)
- o Table of past-run hourly simulated streamflow values
- o Two superimposed past run hydrographs, simulated and observed
- o Table of past run daily simulated and observed hydrograph parameters: precipitation, surface runoff, base flow, and storm runoff
- o Table of total past run simulated and observed peak flow, time of peak, runoff and precipitation
- o Soil moisture storages and parameters at the end of each day of the past run: interflow (IFS), upper zone (UZS), lower zone (SZS), ground water (GWS), baseflow nonlinear recession index (BFNX), seasonal infiltration adjustment multiplier (SIAM) and upper zone capacity (UZC).
- o Table of soil moisture storages and parameters at the end of each day of the past run.

TOWN CREEK

WORSE CASE HOURLY CFS VALUES

NOV

4	AM	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.3	1.0	1.0	1.0	1.0	
	PM	1.0	1.0	1.1	1.9	3.3	6.1	10.4	15.1	26.3	62.9	228.4	480.1	35.5
5	AM	753.8	1027.0	1181.8	1329.0	1453.2	1584.5	1705.7	1792.8	1803.2	1663.9	1499.5	1462.4	
	PM	1447.7	1389.6	1202.4	984.9	800.5	653.0	535.7	443.3	371.1	314.4	270.1	235.5	1079.0
6	AM	208.4	187.1	170.5	157.5	147.3	139.3	132.9	127.9	123.9	120.7	118.2	116.2	
	PM	114.6	113.3	112.2	111.4	110.6	109.9	109.4	108.9	108.6	108.1	107.8	107.4	128.0
7	AM	107.1	106.8	106.4	106.1	105.8	105.5	105.2	105.0	104.3	104.1	103.8	103.5	
	PM	103.3	103.0	102.7	102.5	102.2	101.9	101.6	101.3	101.4	101.1	100.8	100.5	103.6
8	AM	100.2	99.9	99.6	99.3	99.0	98.7	98.4	98.1	97.4	97.1	96.8	96.5	
	PM	96.2	95.9	95.6	95.4	95.1	94.8	94.5	94.2	94.4	94.1	93.8	93.5	96.6

FORECAST CASE HOURLY CFS VALUES

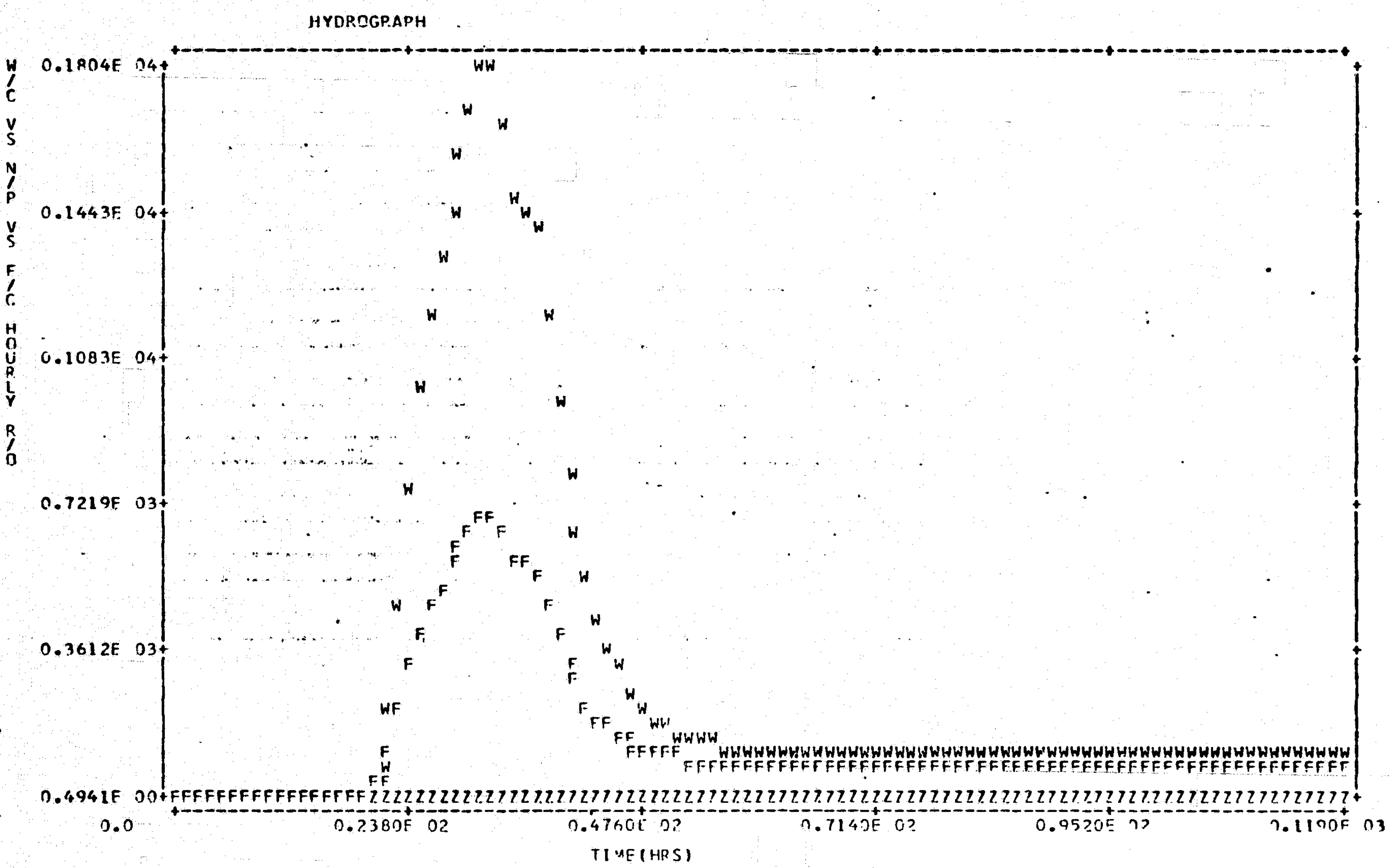
NOV

4	AM	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.3	1.0	1.0	1.0	1.0	
	PM	1.0	1.0	1.0	1.4	2.4	4.4	7.6	11.0	18.8	41.3	109.9	205.7	17.5
5	AM	338.1	409.9	466.3	521.1	566.0	613.5	656.6	686.7	689.0	637.6	575.6	566.6	
	PM	559.8	537.5	468.8	391.0	325.2	272.4	230.2	197.0	171.1	150.6	134.6	122.1	427.4
6	AM	112.2	104.5	98.4	93.7	90.0	87.0	84.6	82.8	81.1	79.9	79.0	78.2	
	PM	77.5	77.0	76.6	76.2	75.9	75.5	75.2	75.0	74.9	74.7	74.4	74.2	82.4
7	AM	74.0	73.8	73.6	73.4	73.2	73.1	72.9	72.7	72.2	72.0	71.8	71.6	
	PM	71.4	71.3	71.1	70.9	70.7	70.5	70.3	70.1	70.3	70.1	69.9	69.7	71.7
8	AM	69.5	69.2	69.0	68.8	68.6	68.4	68.2	68.0	67.4	67.2	67.0	66.8	
	PM	66.6	66.4	66.2	66.0	65.8	65.6	65.4	65.2	65.5	65.3	65.1	64.9	66.9

NO PRECIP HOURLY CFS VALUES

NOV

4	AM	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.3	1.0	1.0	1.0	1.0	
	PM	1.0	1.0	0.9	0.9	0.9	0.9	0.9	0.9	1.3	1.3	1.3	1.3	1.1
5	AM	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.1	1.1	1.1	1.1	
	PM	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.0	1.2	1.2	1.2	1.2	1.2
6	AM	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.0	1.0	1.0	1.0	
	PM	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.1	1.1	1.1	1.1	1.1
7	AM	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	0.7	0.7	0.7	0.7	
	PM	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	1.0	1.0	1.0	1.0	0.9
8	AM	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.5	0.5	0.5	0.5	
	PM	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	1.0	1.0	1.0	1.0	0.8


HYDROGRAPH
W/C VS N/P VS F/C HOURLY R/O
0.1804E 04
0.1443E 04
0.1083E 04
0.7219E 03
0.3612E 03
0.4941E 00
0.0
0.2380E 02
0.4760E 02
0.7140E 02
0.9520E 02
0.1190E 03
TIME(HRS)

TOWN CREEK

TABLE ONE FORECAST RUN

NOV		WORST CASE	NO PRECIP	FORECAST
4	PRECIP	3.204	0.0	2.670
	SUR R/O	0.010	-.000	0.004
	INT FL	0.0	0.0	0.0
	BASE FL	0.003	0.000	0.002
	STM R/O	48.2	1.1	25.3
5	PRECIP	0.552	0.0	0.460
	SUR R/O	0.368	-.000	0.152
	INT FL	0.0	0.0	0.0
	BASE FL	0.035	0.000	0.026
	STM R/O	1528.5	1.2	672.4
6	PRECIP	0.0	0.0	0.0
	SUR R/O	0.007	-.000	0.003
	INT FL	0.0	0.0	0.0
	BASE FL	0.035	0.000	0.026
	STM R/O	158.4	1.1	107.9
7	PRECIP	0.0	0.0	0.0
	SUR R/O	0.000	-.000	0.000
	INT FL	0.0	0.0	0.0
	BASE FL	0.033	0.000	0.024
	STM R/O	124.5	0.9	91.8
8	PRECIP	0.0	0.0	0.0
	SUR R/O	-.000	-.000	-.000
	INT FL	0.0	0.0	0.0
	BASE FL	0.031	0.000	0.023
	STM R/O	116.1	0.8	85.6

TABLE TWO FORECAST RUN

	WORST CASE	NO PRECIP	FORECAST	DIFF	%DIFF
PEAK (CFS)	2572.3	1.4	1102.3	1470.00	*****
PEAK (HR)	33	1	33	0	0.0
R/O (IN)	0.52	0.00	0.26	0.26	*****
PRECIP (IN)	3.756	0.0	3.130	0.626	16.67

ORIGINAL PAGE IS
OF POOR QUALITY

TOWN CREEK

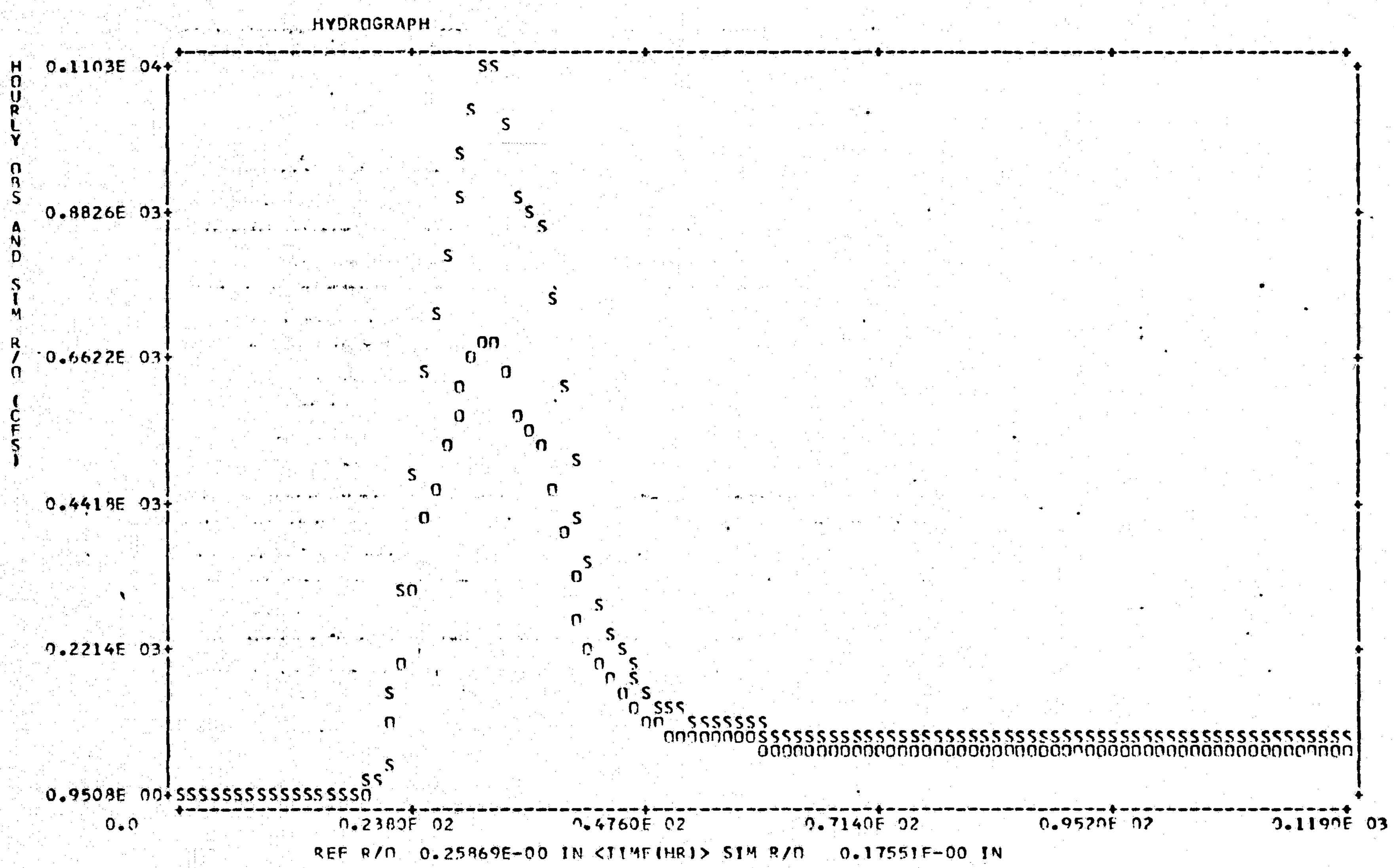
TABLE THREE

NOV		FORECAST RUN WORST CASE	NO PRECIP	FORECAST
4	IFS	0.0	0.0	0.0
	UZS	0.47	0.0	0.45
	LZS	3.32	1.28	3.10
	GWS	0.32	0.00	0.24
	RFNX	0.34	0.02	0.26
	SIAM	0.98	0.98	0.99
	UZC	0.25	0.25	0.25
5	IFS	0.0	0.0	0.0
	UZS	0.40	0.0	0.38
	LZS	3.71	1.26	3.46
	GWS	0.49	0.00	0.36
	RFNX	0.53	0.02	0.39
	SIAM	0.98	0.98	0.99
	UZC	0.25	0.25	0.25
6	IFS	0.0	0.0	0.0
	UZS	0.35	0.0	0.34
	LZS	3.72	1.24	3.49
	GWS	0.47	0.00	0.34
	RFNX	0.52	0.02	0.39
	SIAM	0.97	0.97	0.97
	UZC	0.25	0.25	0.25
7	IFS	0.0	0.0	0.0
	UZS	0.29	0.0	0.28
	LZS	3.73	1.21	3.48
	GWS	0.44	0.00	0.32
	RFNX	0.51	0.02	0.38
	SIAM	0.97	0.97	0.97
	UZC	0.25	0.25	0.25
8	IFS	0.0	0.0	0.0
	UZS	0.23	0.0	0.22
	LZS	3.73	1.18	3.48
	GWS	0.41	0.00	0.30
	RFNX	0.49	0.02	0.37
	SIAM	0.97	0.97	0.97
	UZC	0.25	0.25	0.25

ORIGINAL PAGE IS
OF POOR QUALITY

PAST RUN HOURLY CFS VALUE

NOV														
4	AM	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.3	1.0	1.0	1.0	1.0	
	PM	1.0	1.0	1.1	1.7	2.9	5.6	9.8	14.1	24.6	54.7	160.9	315.0	25.3
5	AM	480.6	645.2	736.7	824.5	897.9	975.6	1046.9	1097.5	1102.3	1017.9	914.2	899.1	
	PM	890.0	854.8	742.5	613.3	504.4	417.4	348.4	294.1	251.7	218.4	192.3	171.9	672.4
6	AM	155.9	143.4	133.5	125.8	119.8	115.0	111.2	108.2	105.7	103.8	102.2	101.0	
	PM	100.0	99.2	98.5	97.9	97.4	96.9	96.5	96.2	96.0	95.6	95.3	95.1	107.9
7	AM	94.8	94.5	94.2	93.9	93.7	93.5	93.2	93.0	92.4	92.2	91.9	91.7	
	PM	91.5	91.2	91.0	90.8	90.5	90.3	90.0	89.7	89.8	89.6	89.3	89.1	91.8
8	AM	89.8	88.5	88.3	88.0	87.7	87.5	87.2	87.0	86.2	86.0	85.7	85.5	
	PM	85.2	85.0	84.7	84.5	84.2	84.0	83.7	83.4	83.7	83.4	83.2	82.9	85.6


HYDROGRAPH
HOURLY OBS AND SIM R/O (CFS)
0.1103E 04
0.8826E 03
0.6622E 03
0.4418E 03
0.2214E 03
0.9508E 00
0.0
0.2380E 02
0.4760E 02
0.7140E 02
0.9520E 02
0.1190E 03
REF R/O 0.25869E-00 IN <TIME(HR)> SIM R/O 0.17551E-00 IN

ORIGINAL PAGE IS OF POOR QUALITY

TOWN CREEK

TABLE ONE NOV	PAST RUN OBSERVED	SIMULATED
4PRECIP	2.670	2.670
SUR R/O		0.004
INT FL		0.0
BASE FL		0.002
STM R/O	17.7	25.3
5PRECIP	0.460	0.460
SUR R/O		0.152
INT FL		0.0
BASE FL		0.026
STM R/O	431.4	672.4
6PRECIP	0.0	0.0
SUR R/O		0.003
INT FL		0.0
BASE FL		0.026
STM R/O	87.1	107.9
7PRECIP	0.0	0.0
SUR R/O		0.000
INT FL		0.0
BASE FL		0.024
STM R/O	72.4	91.8
8PRECIP	0.0	0.0
SUR R/O		-.000
INT FL		0.0
BASE FL		0.023
STM R/O	67.8	85.6

TABLE TWO PAST RUN

	OBSERVED	SIMULATED	DIFF	% DIFF
PEAK (CFS)	68[illegible].7	110[illegible].3	412.5[illegible]	59.8
PEAK (HR)	33	33	0	0.0
R/O (IN)	0.2[illegible]	0.1[illegible]	0.08	47.4
PRECIP IN	3.130	3.130		

TOWN CREEK

TABLE THREE PAST RUN

NOV		SIMULATED
4	IFS	0.0
	UZS	0.45
	LZS	3.10
	GWS	0.24
	BFNX	0.26
	SIAM	0.98
	UZC	0.25
5	IFS	0.0
	UZS	0.38
	LZS	3.46
	GWS	0.36
	BFNX	0.39
	SIAM	0.98
	UZC	0.25
6	IFS	0.0
	UZS	0.34
	LZS	3.48
	GWS	0.34
	BFNX	0.39
	SIAM	0.97
	UZC	0.25
7	IFS	0.0
	UZS	0.28
	LZS	3.48
	GWS	0.32
	BFNX	0.38
	SIAM	0.97
	UZC	0.25
8	IFS	0.0
	UZS	0.22
	LZS	3.48
	GWS	0.30
	BFNX	0.37
	SIAM	0.97
	UZC	0.25

SECTION 8

REFERENCES

1. Crawford, Norman H., and Linsley, R. K. Digital Simulation in Hydrology: Stanford Watershed Model IV. Stanford, California: Stanford University, Department of Civil Engineering, Technical Report No. 39, 1966.

2. Crawford, Norman H., and Linsley, R. K., The Synthesis of Continuous Streamflow Hydrographs on a Digital Computer. Stanford, California; Stanford University, Department of Civil Engineering, Technical Report No. 12, July 1962.

3. Anderson, E. A., and Crawford, N. H. The Synthesis of Continuous Snowmelt Hydrographs on a Digital Computer. Stanford, California: Stanford University, Department of Civil Engineering, Technical Report No. 36, 1964.

4. James L. Douglas. Economic Analysis of Alternative Flood Control Measures Lexington: University of Kentucky, Water Resources Institute, Research Report No. 16, 1968.

5. Ligon, James T., Law, Albert G., and Higgins, Donald H. Evaluation and Application of a Digital Hydrologic Simulation Model. Clemson, South Carolina: Clemson University, Water Resources Research Institute, Report No. 12, November 1969.

6. Linsley, Ray K., Kohler, Max A., and Paulhus, Joseph L. H. Hydrology for Engineers. New York: McGraw-Hill Book Co., 1958.

7. James, L. Douglas. An Evaluation of Relationships between Streamflow Patterns and Watershed Characteristics through the Use of OPSET: A Self-Calibrating Version of the Stanford Watershed Model. Lexington: University of Kentucky, Water Resources Institute, Research Report No. 36, 1970.

8. Liou, Earnest Y. OPSET: Program for Computerized Selection of Watershed Parameter Values for the Stanford Watershed Model. Lexington: University of Kentucky, Water Resources Institute, Research Report No. 34, 1970.

9. Ross, Glendon A. The Stanford Watershed Model: The Correlation of Parameter Values Selected by a Computerized Procedure with Measurable Physical Characteristics of the Watershed. Lexington: University of Kentucky Water Resources Institute, Research Report No. 35, 1970.

10. Chow, Ven Te (ed.). Handbook of Applied Hydrology. New York: McGraw-Hill Book Company, 1964.

11. Striffler, W. D., User Manual for the Colorado State University Version of the Kentucky Watershed Model, Colorado State University, September 1973.

12. Claborn, B. J., and Moore, Walter L. Numerical Simulation of Watershed Hydrology. Austin: University of Texas, Department of Civil Engineering, Technical Report HYD 14-7001, August, 1970.

13. Clark, C. O., "Storage and the Unit Hydrograph," Transaction A.S.C.E., Vol. 110, 1945, pp. 1419-1488.

14. Clarke, K. D., Application of Stanford Watershed Model Concepts to Predict Flood Peaks for Small Drainage Areas. Lexington, Kentucky: Kentucky Department of Highways, Division of Research, HPR-1(3): KYHPR-46-23, 1968.

15. James, L. D., and Thompson, W. O., "Least Squares Estimation of Constants in a Linear Recession Model," Water Resources Research, Vol. 6, August 1970.

16. Leopold, L. B., and Maddock, Jr., T., "The Hydraulic Geometry of Stream Channel and Some PhysiographicImplications," Geological Survey Professional Paper 252, U.S. Government Printing Office, Washington, D.C., 1953.

17. Briggs, D. L., "Application of the Stanford Streamflow Simulation Model to Small Agricultural Watersheds at Coshocton, Ohio," Master's Thesis, Department of Civil Engineering, the Ohio State University, 1969.

18. Anderson, T. W., "Electrical Analog Analysis of Ground Water Depletion in Central Arizona," Geological Survey Professional Paper 1860, Washington, D.C., 1968.

19. Ricci, Vincent T., 1972: The Ohio State University Version of the Stanford Streamflow Simulation Model, Office of Water Resources Research, U. S. Department of the Interior: Part I - Technical Aspects, May 1972; Part III - User's Manual, August 1972.

20. Corps of Engineers, 1956: "Snow Hydrology" North Pacific Division, Corps of Engineers, U. S. Army, Portland, Oregon.

21. Gray, D. M., 1971: "Handbook on Principles of Hydrology," Canadian National Committee for IHD.

22. Leaf, C. F. and Brink, G. E., 1973a: "Computer Simulation of Snowmelt Within a Colorado Subalpine Watershed," USDA Forest Service Research Paper RM-99.

23. Leaf, C. F. and Brink, G. E., 1973b: "A Hydrologic Model of Colorado Subalpine Forest," USDA Forest Service Research Paper RM-107.

24. Leopold, L. B. and Maddock, T., 1953: "The Hydraulic Geometry of Stream Channels and Some Physiographic Implications," U.S. Geologic Survey, Professional Paper 252.

25. Owen, S. M., 1970: "Modifications to the Stanford Streamflow Watershed Model IV to Improve Groundwater Simulation for Stratified Geologic Region," Department of Civil Engineering, Ohio State University (unpublished MS thesis).

26. Hamon, W. R., 1961: "Estimating Potential Evapotranspiration," Proceedings of the A.S.C.E Journal Hydraulics Division 87 (HY3).

APPENDIX A

DICTIONARY OF VARIABLES

DICTIONARY OF VARIABLES
THE NASA-IBM WATERSHED MODEL

ITEM 1 - VARIABLE NAME
ITEM 2 - WHETHER VARIABLE IS REAL, INTEGER, OR LOGICAL
ITEM 3 - VARIBLE DIMENSIONS
ITEM 4 - UNITS
ITEM 5 - DEFINITION OF THE VARIABLE

1	2	3	4	5
ABFSL	R	1	DAY	ACCUMULATED BASE FLOW SEQUENCE LENGTH
ABFV	R	1	IN	ANNUAL BASE FLOW VOLUME
ACRFMI	R	1	-	ACCUMULATED CASES IN ALL RECORDED FLOOD MAGNITUDE INTERVALS
ADBF	R	1	IN	ACCUMULATED DAILY BASE FLOW
ADIF	R	1	IN	ACCUMULATED DAILY INTERFLOW
ADRSP	R	1	CFS	ACCUMULATED SUM OF DRSP
AETX	R	1	IN	ANNUAL EVAPOTRANSPIRATION INDEX
AEX90	R	1	IN	ANTECEDENT EVAPORATION INDEX, DECAY RATE = 0.9
AEX96	R	1	IN	ANTECEDENT EVAPORATION INDEX, DECAY RATE = 0.96
AFSIL	R	1	IN	ANNUAL FOREST SNOW INTERCEPTION LOSS
AHP	R	1	IN	ACCUMULATED HOURLY PRECIPITATION
AIFSL	R	1	DAY	ACCUMULATED INTERFLOW SEQUENCE LENGTH
AIFV	R	1	IN	ANNUAL INTERFLOW VOLUME
AMBER	R	1	IN	ANNUAL MOISTURE BALANCE ERROR
AMBF	R	1	IN	ACCUMULATED DAILY BASE FLOW
AMFSIL	R	1	IN	ACCUMULATED DAILY FOREST SNOW INT5R35PT9CN LOSS
AMIF	R	1	IN	ACCUMULATED DAILY INTERFLOW
AMNET	R	1	IN	ACCUMULATED DAILY NET EVAPOTRANSPIRATION
AMPET	R	1	IN	ACCUMULATED DAILY POTENTIAL EVAPOTRANSPIRATION
AMPREC	R	1	IN	ACCUMULATED DAILY PRECIPITATION
AMRPM	R	1	IN	ACCUMULATED DAILY RAIN PLUS MELT
AMRTF	R	1	CFS	ACCUMULATED DAILY RECORDED TOTAL FLOW
AMSE	R	1	IN	ACCUMULATED DAILY STREAM EVAPORATION
AMSNE	R	1	IN	ACCUMULATED DAILY SNOW EVAPORATION
AMSTF	R	1	CFS	ACCUMULATED DAILY SYNTHESIZED TOTAL FLOW
ANET	R	1	IN	ANNUAL NET EVAPOTRANSPIRATION
AOFV	R	1	IN	ANNUAL OVERLAND FLOW VOLUME
APET	R	1	IN	ANNUAL POTENTIAL EVAPOTRANSPIRATION
APPKP	R	1	-	ACCUMULATED PARAMETER PEAK PRODUCTS
APREC	R	1	IN	ANNUAL PRECIPITATION
AREA	R	1	SQ MI	AREA OF WATERSHED
ARHF	R	1	IN	ACCUMULATED ROUTED HYDROGRAPH FLOW
ARHPF	R	1	CFS	ACCUMULATED RECORDED HYDROGRAPH PEAK FLOWS
ARPM	R	1	IN	ANNUAL RAIN PLUS MELT
ARSTR	R	1	-	ACCUMULATED RATIO OF SYNTHESIZED TO RECORDED FLOWS
ASE	R	1	IN	ANNUAL SNOW EVAPORATION
ASEV	R	1	IN	ANNUAL STREAM EVAPORATION VOLUME
ASM	R	1	IN	ANNUAL SNOWFALL MOISTURE
ASMRG	R	1	IN	ANNUAL SNOWFALL MOISTURE REACHING GROUND
ASRR	R	5,21	CFS	ABSTRACTED SYNTHESIZED ROUTED RUNOFFS
ASRX	R	1	-	AVERAGE VALUE OF SRX
ATF	R	1	SFD	ACCUMULATED TOTAL FLOW
ATFV	R	1	SFD	ANNUAL TOTAL FLOW VOLUME
AVRHPF	R	1	CFS	AVERAGE VALUE OF RHPF
AWSBIT	R	99	-	ACCUMULATOR FOR WATERSHED BITS
BBMIR	R	1	IN/HR	CURRENT BEST ESTIMATE OF BASIC MAXIMUM INFILTRATION RATE
BBUZC	R	1	-	CURRENT BEST ESTIMATE OF BASIC UPPER ZONE STORAGE CAPACITY FACTOR
BBYLZS	R	1	IN	CURRENT BEST ESTIMATE OF BEGINNING OF YEAR LOWER ZONE STORAGE
BDDFSM	R	1	IN/HR	BASIC DEGREE DAY FACTOR FOR SNOW MELT
BETLF	R	1	-	CURRENT BEST ESTIMATE OF EVAPOTRANSPIRATION LOSS FACTOR
BFHRC	R	1	-	BASE FLOW HOURLY RECESSION CONSTANT
BFNHR	R	1	-	BASE FLOW HOURLY NONLINEAR RECESSION ADJUSTMENT FACTOR
BFNLR	R	1	-	BASE FLOW NONLINEAR RECESSION ADJUSTMENT FACTOR
BFNRL	R	1	-	BASE FLOW NONLINEAR RECESSION LOGARITHM
BFNX	R	1	IN	CURRENT VALUE OF BASE FLOW NONLINEAR RECESSION INDEX
BFRC	R	1	-	BASE FLOW RECESSION CONSTANT
BFRL	R	1	-	BASE FLOW RECESSION LOGARITHM
BFSRX	R	1	-	CURRENT BEST ESTIMATE OF FSRX
BISRX	R	1	-	BIG INCREMENTAL STORAGE ROUTING INDEX
BIVF	R	1	-	BASIC INTERFLOW VOLUME FACTOR
BLZC	R	1	IN	CURRENT BEST ESTIMATE OF LOWER ZONE STORAGE CAPACITY
BMIR	R	1	IN/HR	BASIC MAXIMUM INFILTRATION RATE WITHIN WATERSHED
BSIAC	R	1	-	CURRENT BEST ESTIMATE OF SEASONAL INFILTRATION ADJUSTMENT FACTOR
BSUZC	R	1	-	CURRENT BEST ESTIMATE OF SEASONAL UPPER ZONE STORAGE CAPACITY FACTOR
BTRI	R	99	-	BASE TIME ROUTING INCREMENTS

BUZC	R	1	-	BASIC UPPER ZONE STORAGE CAPACITY FACTOR
BYGWS	R	1	IN	BEGINNING OF YEAR GROUNDWATER STORAGE
BYIFS	R	1	IN	BEGINNING OF YEAR INTERFLOW STORAGE
BYLZS	R	1	IN	BEGINNING OF YEAR LOWER ZONE STORAGE
BYUZS	R	1	IN	BEGINNING OF YEAR UPPER ZONE STORAGE
CBF	R	1	IN/HR	CURRENT BASE FLOW
CCRFMI	R	1	-	CASES IN CURRENT RECORDED FLOW MAGNITUDE INTERVAL
CDSDR	I	1	-	CURRENT DAY FOR WHICH STORM DETAILS REQUESTED
CFBF	R	1	CFS	CURRENT HYDROGRAPH BASE FLOW
CFCAP	R	1	CFS	CHANNEL CAPACITY - INDEXED TO BASIN OUTLET
CHPV	R	1	-	CURRENT HYDROGRAPH PARAMETER VALUE
CIVM	R	1	-	CURRENT INTERFLOW VOLUME MULTIPLIER
CMIR	R	1	IN	CURRENT MAXIMUM INFILTRATION RATE DURING PERIOD
CN	I	1	-	1 = A. M., 2 = P. M.
CONOPT	I	15	-	CONTROL OPTION
CONOP2	I	1	-	SECOND CONTROL OPTION
CRFMI	R	22	-	CASES RECORDED IN FLOW MAGNITUDE INTERVAL
CRSBBF	R	1	CFS	CURRENT RECESSION SEQUENCE BEGINNING BASE FLOW
CRSBIF	R	1	CFS	CURRENT RECESSION SEQUENCE BEGINNING INTERFLOW
CRSBTF	R	1	CFS	CURRENT RECESSION SEQUENCE BEGINNING TOTAL FLOW
CRSTF	R	50	CFS	CURRENT RECESSION SEQUENCE TOTAL FLOWS
CRSOBF	R	1	CFS	CURRENT RECESSION SEQUENCE BASE FLOW ON DAY ZERO
CRSOIF	R	1	CFS	CURRENT RECESSION SEQUENCE INTERFLOW ON DAY ZERO
CSRX	R	1	-	CHANNEL STORAGE ROUTING INDEX
CTRI	R	99	-	CURRENT TIME ROUTING INCREMENTS
DATE	I	1	-	CURRENT DAY OF THE MONTH
DAY	I	1	-	CURRENT DAY OF THE YEAR
DBFRC	R	1	-	DOUBLE PRECISION BFRC
DDIW	R	366	CFS	DATED DIVERSIONS INTO WATERSHED
DFCC	R	1	-	DAILY FLOW CORRELATION COEFFICIENT
DIFRC	R	1	-	DOUBLE PRECISION IFRC
DIV	R	1	CFS	DIVERSION INTO BASIN, MEAN DAILY FLOW
DMNT	R	366	DEGF	DATED MINIMUM TEMPERATURE
DMXT	R	366	DEGF	DATED MAXIMUM TEMPERATURE
DNFS	R	1	-	DENSITY OF NEW FALLEN SNOW
DPET	R	366	IN	DATED POTENTIAL EVAPOTRANSPIRATION
DPSE	R	366	IN	DATED POTENTIAL SNOW EVAPORATION
DPY	I	1	-	DAYS PER YEAR
DRAF	R	1	CFS	DIFFERENCE BETWEEN RECORDED AND AVERAGE FLOW
DRGPM	R	366	-	DATED RECORDING GAGE PRECIPITATION MULTIPLIER
DRHP	R	366,24	IN	DATED RECORDED HOURLY PRECIPITATION
DRSF	R	366	CFS	DATED RECORDED STREAMFLOW
DRSGP	R	366	IN	DATED RECORDED STORAGE GAGE PRECIPITATION
DRSP	R	1	CFS	DIFFERENCE BETWEEN RECORDED AND SYNTHESIZED HYDROGRAPH PEAKS
DSAF	R	1	CFS	DIFFERENCE BETWEEN SYNTHESIZED AND AVERAGE FLOW
DSMGH	R	1	IN	RATE OF DAILY SNOWMELT FROM GROUND HEAT
DSSF	R	366	CFS	DATED SYNTHESIZED STREAMFLOW
EDATE	I	1		ENDING DATE OF FORECAST
EDLZS	R	366	IN	END OF DAY VALUES OF LZS
EHSGD	I	1	-	ENDING HOUR OF STORAGE GAGE DAY
EHSGDF	R	1	-	ENDING HOUR OF STORAGE GAGE DAY - FLOATING POINT
EID	R	1	-	EXPONENT OF INFILTRATION RATE DECAY WITH INCREASED SOIL MOISTURE CONENT
ELDIF	R	1	1000FT	ELEVATION DIFFERENCE BETWEEN BASE THERMOMETER AND BASIN MEAN ELEVATION
EMAET	R	1	IN	ESTIMATED MAXIMUM ANNUAL EVAPOTRANSPIRATION
EMATF	R	13	SFD	END OF MONTH ACCUMULATED TOTAL FLOWS
EMBFNX	R	15,3	IN	DAILY BASE FLOW NONLINEAR RECESSION INDEX
EMFDP	R	1	-	EXTREME MONTHLY FLOW DEVIATION PARAMETER
EMGWS	R	15,3	IN	DAILY GROUND WATER STORAGE
EMIFS	R	15,3	IN	DAILY INTERFLOW STORAGE
EMLZS	R	15,3	IN	DAILY LOWER ZONE STORAGE
EMONTH	I	1		ENDING MONTH OF FORECAST
EMSIAM	R	15,3	IN	DAILY SEASONAL INFILTRATION ADJUSTMENT MULT.
EMUZC	R	15,3	IN	DAILY UPPER ZONE STORAGE CAPACITY
EMUZS	R	15,3	IN	DAILY UPPER ZONE STORAGE
EPAET	R	1	IN	ESTIMATED POTENTIAL ANNUAL EVAPOTRANSPIRATION
EPCM	R	12	-	EVAPORATION PAN COEFFICIENT FOR MONTH
EPS	R	1	-	MAXIMUM REQUIRED ESTIMATING TOLERANCE
EQD	R	1	IN	EQUILIBRIUM DEPTH OF OVERLAND FLOW
EQDF	R	1	-	EQUILIBRIUM DEPTH FACTOR FOR OVERLAND FLOW
EQDFIS	R	1	-	EQUILIBRIUM DEPTH FACTOR FOR OVERLAND FLOW, IMPERVIOUS SURFACES
EQDIS	R	1	IN	EQUILIBRIUM DEPTH OF OVERLAND FLOW IMPERVIOUS SURFACES
ERR	R	1	CFS	DIFFERENCE BETWEEN RECORDED AND SYNTHESIZED DATED STREAMFLOW
ETIBF	R	1	CFS	ERROR TABLE INTERVAL BOUNDARY FLOODS
ETLF	R	1	-	EVAPOTRANSPIRATION LOSS FACTOR
EXCPV	R	1	-	EXPONENT OF FLOW PROPORTIONAL TO VELOCITY
FBUZC	R	1	-	ADJUSTMENT FACTOR FOR BUZC
FCNTRI	R	1	-	FLOATING POINT CHANGE IN NUMBER OF TIME ROUTING INCREMENTS
FDAY	R	1	-	FLOATING POINT CURRENT DAY OF THE YEAR
FDPY	R	1	-	FLOATING POINT DAYS PER YEAR
FDSC	R	1	-	FIRST DIFFERENTIAL OF SINE CURVE MAGNITUDE
FETLF	R	1	-	ADJUSTMENT FACTOR FOR ETLF
FFOR	R	1	-	FRACTION OF THE WATERSHED BEING FOREST
FFSI	R	1	-	FRACTION OF SNOW ON FOREST INTERCEPTED
FHPP	R	1	-	FRACTIONAL HOUR PER PERIOD
FIMP	R	1	-	FRACTION OF THE WATERSHED BEING IMPERVIOUS
FIRR	R	15	-	FRACTION OF INCOMING RADIATION REFLECTED BY SNOW SURFACE AS A FUNCTION OF AGE

FKREMI	R	1	-	FLOATING POINT VALUE OF KREMI
FLZC	R	1	-	ADJUSTMENT FACTOR FOR LZC
FMR	R	1	-	FRACTION OF MOISTURE RETENTION
FMXTRI	R	1	-	FLOATING POINT MAXIMUM NUMBER OF TIME ROUTING INCREMENTS
FNBTRI	R	1	-	FLOATING POINT NUMBER OF BASIC TIME ROUTING INCREMENTS
FNCTRH	R	1	-	FLOATING POINT NUMBER OF CURRENT TIME ROUTING HOURS
FNCFM	R	1	-	FLOATING POINT NUMBER OF OVERLAND FLOW MONTHS
FNPTRI	R	1	-	FLOATING POINT NUMBER OF PREVIOUS TIME ROUTING INCREMENTS
FNRHP	R	1	-	FLOATING POINT NUMBER OF RECORDED HYDROGRAPH PEAKS
FNSTRI	R	1	-	FLOATING POINT NUMBER OF SUBSEQUENT TIME ROUTING INCREMENTS
FNTRI	R	1	-	FLOATING POINT NUMBER OF TIME ROUTING INCREMENTS
FPER	R	1	-	FRACTION OF THE WATERSHED BEING PERVIOUS
FRERS	R	1	CFS	FLOW RISE ENDING RECESSION SEQUENCE
FSIAC	R	1	-	ADJUSTMENT FACTOR FOR SIAC
FSIL	R	1	IN	HOURLY FOREST SNOW INTERCEPTION LOSS
FSRX	R	1	-	FLOOD PLAIN STORAGE ROUTING INDEX
FSUZC	R	1	-	ADJUSTMENT FACTOR FOR SUZC
FTA	R	1	-	FACTOR FOR ESTIMATING DIURNAL TEMPERATURE VARIATION BASED ON SINE CURVE
FTX	R	1	-	FALL TROUBLE INDEX
FWTR	R	1	-	FRACTION OF THE WATERSHED BEING WATER
GWET	R	1	IN	CURRENT HOURLY GROUNDWATER EVAPOTRANSPIRATION
GWETF	R	1	-	GROUNDWATER EVAPOTRANSPIRATION FACTOR
GWS	R	1	IN	CURRENT GROUNDWATER STORAGE
HBF	R	5	CFS	HYDROGRAPH BASE FLOW
HBFM	R	1	-	HYDROGRAPH BASE FLOW MULTIPLIER
HNTRI	I	5	-	HYDROGRAPH NUMBER OF TIME ROUTING INCREMENTS
HOUR	I	1	-	CURRENT HOUR OF THE DAY
HOURF	R	1	HR	CURRENT HOUR OF THE DAY, FLOATING POINT
HRF	I	1	-	FIRST HOUR OF LOOP
HRL	I	1	-	LAST HOUR OF LOOP
HSE	R	1	IN	CURRENT HOURLY STREAM EVAPORATION
HSF	R	1	IN	HOURLY SNOWFALL
HSFRG	R	1	IN	HOURLY SNOWFALL REACHING GROUND
HSM	R	1	IN	HOURLY SNOWMELT RATE
HSRX	R	5	-	HYDROGRAPH STORAGE ROUTING INDEX
HTH	R	1	-	HOURS INTO DAY WHEN HYDROGRAPH STOPS OR STARTS
IBTPR	I	1	HR	TIME FROM BEGINNING OF SAVED RUNOFF TO RECORDED HYDROGRAPH PEAK
IBTPS	I	1	-	TIME FROM BEGINNING OF SAVED RUNOFF TO SYNTHESIZED HYDROGRAPH PEAK
IDAY1	I	1	-	INDEX TO 10-DAY PERIOD
IDAY2	I	1	-	INDEX WITHIN 10-DAY PERIOD
IDYB	I	5	-	DAY OF ROUTING HYDROGRAPH BEGINNING
IDYE	I	5	-	DAY OF ROUTING HYDROGRAPH ENDING
IFPRC	R	1	-	INTERFLOW PERIOD RECESSION CONSTANT
IFRC	R	1	-	INTERFLOW RECESSION CONSTANT
IFRL	R	1	-	INTERFLOW RECESSION LOGARITHM
IFS	R	1	IN	INTERFLOW STORAGE
IFT	I	1	-	INDICATOR OF FALL TROUBLE (SKIP FIRST RECESSION IN EVALUATION OF AMIR)
IHRB	I	5	-	HOUR OF DAY OF ROUTING HYDROGRAPH BEGINNING
IHRE	I	5	-	HOUR OF DAY OF ROUTING HYDROGRAPH ENDING
INHPT	I	1	HR	INTEGER NUMBER OF HOURS BETWEEN HYDROGRAPH PRINTING POINTS
IPPH	I	1	-	INTEGER PERIODS PER HOUR
IPTE	I	1	HR	TIME FROM PEAK OF RECORDED HYDROGRAPH TO END OF SAVED RUNOFF
IRSTRT	I	1		RESTART OPTION-1=RESTART,0=NO RESTART
ISGRD	I	1	-	CURRENT STORAGE GAGE RAINFALL DAY
IWBG	I	1	-	INDEX NUMBER OF WEATHER BUREAU PRECIPITATION GAGE
KAA	I	1	-	COUNTER OF APPROPRIATE ELEMENT FROM ALBEDO ARRAY
KAAC	I	1	-	PRECEDING VALUE OF KAA
KAFH	I	1	-	COUNTER FOR ABSTRACTED FLOW HYDROGRAPH
KAHP	I	1	-	COUNTER FOR ABSTRACTING HYDROGRAPH POINTS
KBRC	I	1	-	COUNTER OF ROUGH CYCLES SINCE BEST ONE
KB1-7	I	1	-	COUNTERS FOR COMBINING WATERSHED BITS
KDBCK	I	1	-	BACKWARD DAY COUNTER
KDFOR	I	1	-	FORWARD DAY COUNTER
KDRS	I	1	-	COUNTER OF CURRENT DAY IN RECESSION SEQUENCE
KDY	I	1	-	COUNTER FOR DAY
KFFC	I	1	-	COUNTER EQUALLING ONE ON FIRST FINE ADJUSTMENT CYCLE
KHBCK	I	1	-	BACKWARD HOUR COUNTER
KHFOR	I	1	-	FORWARD HOUR COUNTER
KHCUR	I	1	-	COUNTER FOR HOUR OF DAY
KHPT	I	1	-	COUNTER OF CURRENT HYDROGRAPH POINT
KHYD	I	1	-	COUNTER SPECIFYING CURRENT HYDROGRAPH
KH1-3	I	1	-	COUNTERS FOR FIXING HYDROGRAPH ROUTING PARAMETERS
KIA	I	1	-	COUNTER FOR INITIALIZING ARRAYS
KISRX	I	1	-	COUNTER FOR INCREMENTING STORAGE ROUTING INDEX
KLCCA	I	1	-	COUNTER FOR LIMITING NUMBER OF CHANNEL CAPACITY ADJUSTMENTS
KMC	I	1	-	COUNTER INDEXING MONTH OF THE YEAR
KM1-6	I	1	-	MONTH COUNTERS
KPA	I	1	-	COUNTER DESIGNATING PARAMETER TO BE AVERAGED
KPCH	I	1	-	COUNTED POINTS IN CURRENT HYDROGRAPH
KPRD	I	1	-	COUNTER FOR PERIOD
KPSH	I	5	-	COUNTED POINTS IN SUBSCRIPTED HYDROGRAPH
KRC	I	1	-	COUNTER OF CURRENT ROUGH CYCLE
KRD	I	1	-	COUNTER FOR READING DATA ARRAYS
KREMI	I	1	-	COUNTER FOR RECORDED FLOW MAGNITUDE INTERVAL
KRHP	I	1	-	COUNTER FOR RECORDED HYDROGRAPH PEAKS
KRIA	I	1	-	COUNTER OF APPROPRIATE ELEMENT FROM RADIATION INCIDENCE ARRAY

ORIGINAL PAGE IS
OF POOR QUALITY

KRS	I	1	-	COUNTER FOR RECESSION SEQUENCE NUMBER
KSD	I	1	-	COUNTER FOR RECESSION SEQUENCE DAYS
KSC	I	1	-	COUNTER FOR REVISED SEQUENCES
KTA	I	1	-	COUNTER FOR TITLE ARRAY
KTRI	I	1	-	COUNTER FOR TIME ROUTING INCREMENTS
KT20	I	1	-	COUNTER FOR TOP 20 VALUES
KWD	I	1	-	COUNTER FOR WRITING DATA ARRAYS
KWSM	I	1	-	COUNTER OF WET SUMMER MONTHS
K1AH	I	1	-	COUNTER FOR FIRST ACCEPTED HYDROGRAPH
K2AH	I	1	-	COUNTER FOR SECOND ACCEPTED HYDROGRAPH
LBFC	L	20	-	LOGICAL VARIABLE SET TRUE WHERE BASE FLOW ONLY ENCOUNTERED
LBMIR	L	1	-	LOGICAL VARIABLE SET TRUE WHEN EXERCIZING SUBSTITUTE APPROACH FOR EVALUATING BMIR
LBUZC	L	1	-	LOGICAL VARIABLE SET TRUE WHEN EXERCIZING SUBSTITUTE APPROACH FOR EVALUATING BUZC
LDAY	I	1	-	LAST DAY OF YEAR
LETLF	L	1	-	LOGICAL VARIABLE SET TRUE WHEN EXERCIZING SUBSTITUTE APPROACH FOR EVALUATING ETLF
LHOUR	I	1	-	LAST HOUR OF DAY
LLZC	L	1	-	LOGICAL VARIABLE SET TRUE WHEN EXERCIZING SUBSTITUTE APPROACH FOR EVALUATING LZC
LNIBRS	I	1	-	LAST VALUE OF NIBRS
LNPR	L	1	-	LOGICAL VARIABLE SET TRUE FOR NONEQUAL PERIOD RAINFALL
LNTRI	I	1	-	LAST NUMBER OF TIME ROUTING INCREMENTS
LRC	L	1	-	LOGICAL VARIABLE SET TRUE DURING ROUGH ADJUSTMENT CYCLES
LSHA	L	5	-	LOGICAL VARIABLE KEPT TRUE WHILE SYNTHESIZED HYDROGRAPH IS ACCEPTED FOR COMPARISION WITH RECORDED HYDROGRAPH
LSHFT	L	1	-	LOGICAL VARIABLE SET TRUE WHILE SHIFTING THE NUMBER OF TIME ROUTING INCREMENTS
LSHP	L	1	-	LOGICAL VARIABLE SET TRUE DURING STORM HYDROGRAPH PERIODS
LZC	R	1	IN	LOWER ZONE STORAGE CAPACITY
LZRX	R	1	-	LOWER ZONE MOISTURE RETENTION INDEX
LZS	R	1	IN	CURRENT LOWER ZONE STORAGE
LZSR	R	1	-	CURRENT LOWER ZONE STORAGE RATIO (LZS/LZC)
MBDS	I	1	-	MONTH BEGINNING DRY SEASON
MBWS	I	1	-	MONTH BEGINNING WET SEASON
MDAY	I	1	-	DAY OF YEAR OF LAST DAY OF PREVIOUS MONTH
MEDCY	I	12	-	MONTH END DATES - CALENDAR YEAR
MEDWY	I	12	-	MONTH END DATES - WATER YEAR
MFDP	R	12	-	MONTHLY FLOW DEVIATION PARAMETER
MHSM	R	1	IN	MINIMUM HOURLY SNOWMELT RATE
MHTP	I	1	-	MULTIPLIER CONVERTING FROM HOURS TO PERIODS
MNDRS	I	1	-	MAXIMUM NUMBER OF DAYS IN RECESSION SEQUENCE
MNRC	I	1	-	MINIMUM NUMBER OF ROUGH CYCLES
MNRD	R	1	-	MEAN ANNUAL NUMBER OF RAINY DAYS
MNTRI	I	1	-	MINIMUM NUMBER OF TIME ROUTING INCREMENTS
MNX	I	1	-	MONTH INDEX
MONTH	I	1	-	CURRENT MONTH OF THE YEAR
MONTH1	I	1	-	COUNTER FOR BEGINNING MONTH
MRNSM	R	1	IN	MAXIMUM RATE OF NEGATIVE SNOWMELT (SNOW CHILLING)
MRSL	I	1	DAY	MINIMUM RECESSION SEQUENCE LENGTH
MSBDIC	I	1		EBCDIC VALUE OF BEGINNING MONTH
MXA	I	12	-	MONTH INDEX ARRAY (SPECIFYING MONTHS USED IN PARAMETER ADJUSTMENT)
MXTRH	I	1	-	MAXIMUM NUMBER OF TIME ROUTING HOURS
MXTRI	I	1	-	MAXIMUM NUMBER OF TIME ROUTING INCREMENTS
M1R	I	1	-	MONTH WITH MOST RUNOFF
M1SP	I	1	-	MONTH WITH MOST SUMMER PRECIPITATION
M11	I	1	-	SET AT 11 IF AUGUST IS A BASE FLOW MONTH
M12	I	1	-	SET AT 12 IF SEPTEMBER IS A BASE FLOW MONTH
M2R	I	1	-	MONTH WITH SECOND MOST RUNOFF
M2SP	I	1	-	MONTH WITH SECOND MOST SUMMER PRECIPITATION
NATRH	I	1	-	NUMBER OF ANTICIPATED TIME ROUTING HOURS
NBTRI	I	1	-	NUMBER OF BASE TIME ROUTING INCREMENTS
NCSTRI	I	1	-	NUMBER OF CURRENT TIME ROUTING INCREMENTS DURING SHIFTING
NCTRH	I	1	-	NUMBER OF CURRENT TIME ROUTING HOURS
NCTRI	I	1	-	NUMBER OF CURRENT TIME ROUTING INCREMENTS
NDAY	I	1	-	NEXT DAY OF YEAR
NDRS	I	2C	DAY	NUMBER OF DAYS IN RECESSION SEQUENCE
NDRSC	I	1	DAY	NUMBER OF DAYS IN CURRENT RECESSION SEQUENCE
NDRSC1	I	1	DAY	NUMBER OF DAYS IN CURRENT RECESSION SEQUENCE LESS 1
NDRSC2	I	1	DAY	NUMBER OF DAYS IN CURRENT RECESSION SEQUENCE LESS 2
NDSDP	I	1	DAY	NUMBER OF DAYS FOR WHICH STORM DETAILS HAVE ALREADY BEEN PRINTED
NDSDR	I	1	DAY	NUMBER OF DAYS FOR WHICH STORM DETAILS REQUESTED
NFRHA	I	1	-	NUMBER OF FIRST RAINFALL HOUR ADJUSTED, PREVIOUS DAY
		1	-	NUMBER OF FIRST TRIP TO BE RUN FOR A GIVEN SECTION YEAS
NIRTS	I	1	-	NUMBER OF TIME ROUTING INTERVALS FROM RECORDED TO SYNTHESIZED PEAK
NLTR	I	1	-	NUMBER OF LAST TRIP TO BE RUN FOR A GIVEN STATION YEAR
NNSTRI	I	1	-	NUMBER OF NEXT TIME ROUTING INCREMENT DURING SHIFTING
NOFM	I	1	-	NUMBER OF OVERLAND FLOW MONTHS
NORHP	I	1	-	NUMBER OF ORIGINAL RECORDED HYDROGRAPH PEAKS
NPPTS	I	1		NO OF OBSERVED HOURLY STORM SAMPLES
NRHA	I	1	-	NUMBER OF RAINFALL HOURS ADJUSTED, CURRENT DAY
NRHP	I	1	-	NUMBER OF RECORDED HYDROGRAPH PEAKS
NRHP1	I	1	-	NUMBER OF RECORDED HYDROGRAPH PEAKS LESS ONE

ORIGINAL PAGE IS OF POOR QUALITY

NRS	I	1	-	NUMBER OF RECESSION SEQUENCES
NRTRI	I	1	-	NUMBER OF TIME ROUTING INCREMENTS REMAINING TO BE ROUTED
NSGRD	I	1	-	NUMBER OF STORAGE GAGE RAINFALL DAYS
NSYC	I	1	-	NUMBER OF STATION YEAR, CURRENT ONE BEING RUN
NSYT	I	1	-	NUMBER OF STATION YEARS, TOTAL INCLUDED IN A GIVEN JOB
NTRS	I	1	-	NUMBER OF TENTATIVE RECESSION SEQUENCES
NYEAR	I	1		BEGINNING OF WATER YEAR
OCT1BF	R	1	-	OCTOBER FIRST BASE FLOW
OFMN	R	1	-	OVERLAND FLOW MANNING'S N
OFMNIS	R	1	-	OVERLAND FLOW MANNING'S N , IMPERVIOUS SURFACES
OFR	R	1	IN	CURRENT OVERLAND FLOW RUNOFF
OFRF	R	1	-	OVERLAND FLOW ROUTING FACTOR
OFRFIS	R	1	-	OVERLAND FLOW ROUTING FACTOR, IMPERVIOUS SURFACES
OFRIS	R	1	IN	CURRENT OVERLAND FLOW RUNOFF, IMPERVIOUS SURFACES
OFS	R	1	IN	OVERLAND FLOW STORAGE
OFSL	R	1	FT	OVERLAND FLOW SURFACE LENGTH
OFSS	R	1	-	OVERLAND FLOW SURFACE SLOPE
OFUS	R	1	IN	CURRENT OVERLAND FLOW UNROUTED STORAGE
OFUSIS	R	1	IN	CURRENT OVERLAND FLOW UNROUTED STORAGE, IMPERVIOUS SURFACES
PBIVF	R	1	-	PREVIOUS VALUE OF BIVF
PBMIR	R	1	IN/HR	PREVIOUS VALUE OF BMIR
PBUZC	R	1	-	PREVIOUS ESTIMATE OF BASIC UPPER ZONE STORAGE CAPACITY FACTOR
PDATE	I	1		BEGINNING DATE OF FORECAST
PDAY	I	1	-	PREVIOUS DAY OF THE YEAR
PEAI	R	1	IN	PRECIPITION EXCESS AFTER INFILTRATION
PEAKS	I	1	CFS	PEAK OF SIMULATED STORM
PEBI	R	1	IN	PRECIPITATION EXCESS, BEFORE INFILTRATION
PEIS	R	1	IN	PRECIPITATION EXCESS ON IMPERVIOUS SURFACES
PEP	R	1	IN	PRECIPITATION ESTIMATED FOR PERIOD
PET	R	1	IN	CURRENT DAILY POTENTIAL EVAPOTRANSPIRATION
PETLF	R	1	-	PREVIOUS ESTIMATE OF EVAPOTRANSPIRATION LOSS FACTOR
PETU	R	1	IN	UNADJUSTED CURRENT DAILY POTENTIAL EVAPOTRANSPIRATION
PE4P	R	4	IN	PRECEPITATION ESTIMATES FOR 4 PERIODS
PGW	R	1	IN	PERCOLATION TO GROUND WATER
PHRO	I	1	HR	HOUR OF OBSERVED STORM PEAK
PHRS	I	1	HR	HOUR OF SIMULATED STORM PEAK
PLZC	R	1	IN	PREVIOUS ESTIMATE OF LZC
PLZS	R	1	IN	PERCOLATION TO LOWER ZONE STORAGE
PMEIFS	R	1	IN	PERIOD MOISTURE ENTERING INTERFLOW STORAGE
PMELZS	R	1	IN	PERIOD MOISTURE ENTERING LOWER ZONE STORAGE
PMEOFS	R	1	IN	PERIOD MOISTURE ENTERING OVERLAND FLOW STORAGE
PMEUZS	R	1	IN	PERIOD MOISTURE ENTERING UPPER ZONE STORAGE
PMONTH	I	1	-	BEGINNING MONTH OF FORECAST
PPH	R	1	-	PERIODS PER HOUR
PPI	R	1	IN	PRECIPITATION PASSING INTERCEPTION
PRD	I	1	-	CURRENT PERIOD OF THE HOUR
PRDF	R	1	-	CURRENT PERIOD OF THE HOUR-FLOATING POINT
PRH	R	1	IN	PRECIPITATION RECORDED FOR HOUR
PRLH	R	1	IN	PRECIPITATION RECORDED FOR LAST HOUR
PRM1	R	1	IN	PRECIPITATION DURING WETTEST MONTH
PRM2	R	1	IN	PRECIPITATION DURING SECOND WETTEST MONTH
PRNH	R	1	IN	PRECEPITATION RECORDED FOR NEXT HOUR
PSIAC	R	1	-	PREVIOUS ESTIMATE OF SEASONAL INFILTRATION ADJUSTMENT
PSUZC	R	1	-	PREVIOUS ESTIMAED OF SEASONAL UPPER ZONE STORAGE CAPACITY FACTOR
PXCSA	R	1	IN	PRECIPITATION INDEX FOR CHANGING SNOW ALBEDO
Q	R	337	CFS	OBSERVED HOURLY STORM SAMPLES
QMAX	R	1	CFS	PEAK OF OBSERVED STORM
QMIN	R	1	CFS	MINIMUM OF OBSERVED STORM
QCUT	R	15	CFS	MEAN DAILY OBSERVED STORM SAMPLES
RA	R	1	-	RECESSION ALPHA
RAA	R	1	IN	RAINFALL ADJUSTMENT ADDITION
RADF	R	1	CFS	RECORDED AVERAGE DAILY FLOW
RAM	R	1	-	RAINFALL ADJUSTMENT MULTIPLIER
RATFV	R	1	SFD	RECORDED ANNUAL TOTAL FLOW VOLUME
RA1-6	R	1	-	REGRESSION ACCUMULATORS
RB	R	1	-	RECESSION BETA
RBF	R	1	CFS	RECORDED BASE FLOW
RD	R	1	-	RECESSION DISCRIMINANT
RDPT	R	1	IN	RECORDED DAILY PRECIPITATION TOTAL
RFRISE	R	1	IN	RECORDED FLOW RISE
RGPM	R	1	-	RECORDING GAGE PRECIPITATION MULTIPLIER
RGPMB	R	1	-	RECORDING GAGE PRECIPITATION MULTIPLIER - BASIC
RHFMC	R	1	IN	ROUTED HYDROGRAPH FLOW AT MINIMUM CUTOFF
RHFO	R	1	IN	PRECEDING ROUTED HYDROGRAPH FLOW
RHFI	R	1	IN	CURRENT ROUTED HYDROGRAPH FLOW (EXCLUDING BASE FLOW)
RHPD	I	5	-	RECORDED HYDROGRAPH PEAK DAY
RHPF	R	5	CFS	RECORDED HYDROGRAPH PEAK FLOW
RHPH	I	5	-	RECORDED HYDROGRAPH PEAK HOUR
RICD	R	1	-	RADIATION INCIDENCE FOR THE CURRENT DAY
RICY	R	37	-	RADIATION INCIDENCE OVER THE CALENDAR YEAR
RIF	R	1	CFS	RECORDED INTERFLOW
RINT	R	1	-	REGRESSION INTERCEPT
RMFX	R	1	-	RECORDED MONTHLY FLOW INDEX
RMPF	R	1	CFS	REQUESTED MINIMUM DAILY PEAK FLOW TO BE PRINTED
RMWR	R	1	IN	RAINFALL MAXIMUM WITHOUT RUNOFF

ORIGINAL PAGE IS OF POOR QUALITY

RSBBF	R	20	CFS	ESTIMATED BASE FLOW AT BEGINNING OF RECESSION SEQUENCE
RSBD	I	20	-	RECESSION SEQUENCE BEGINNING DAY
RSBFRC	R	20	-	RECESSION SEQUENCE BASE FLOW RECESSION CONSTANT
RSBIF	R	20	CFS	ESTIMATED INTERFLOW AT BEGINNING OF RECESSION SEQUENCE
RSFM	R	1	CFS	RECESSION SEQUENCE FLOW MINIMUM
RSFN	R	1	CFS	RECORDED STREAMFLOW ON NEW DAY
RSF1	R	1	CFS	RECORDED STREAMFLOW ON DAY 1
RSF2	R	1	CFS	RECORDED STREAMFLOW ON DAY 2
RSIFRC	R	20	-	RECESSION SEQUENCE INTERFLOW RECESSION CONSTANT
RSL	I	1	DAY	CURRENT RECESSION SEQUENCE LENGTH
RSLP	R	1	-	REGRESSION SLOPE
RSPTF	R	1	IN	ROUTED SYNTHESIZED PERIOD TOTAL FLOW
RSTF	R	50,20	CFS	RECESSION SEQUENCE TOTAL FLOWS
RSTR	R	1	-	RATIO OF SYNTHESIZED TO RECORDED FLOW
RWRAIN	R	1	IN	RECORDED WATERSHED RAINFALL
SADF	R	1	CFS	SYNTHESIZED AVERAGE DAILY FLOW
SARAX	R	1	IN	SNOW ALBEDO RAINFALL AGING INDEX
SASFX	R	1	IN	SNOW ALBEDO SNOWFALL FRESHENING INDEX
SATFV	R	1	SFD	SYNTHESIZED ANNUAL TOTAL FLOW VOLUME
SATFVI	R	1	IN	SYNTHESIZED ANNUAL TOTAL FLOW VOLUME IN INCHES
SATRI	R	99	-	SHIFT ADJUSTMENTS FOR TIME ROUTING INCREMENTS
SAX	R	1	-	SNOW ALBEDO INDEX
SBF	R	1	CFS	SYNTHESIZED BASE FLOW
SBFRS	R	3,20	CFS	SYNTHESIZED BASE FLOW DURING THE FIRST THREE DAYS OF EACH RECESSION SEQUENCE
SDEPTH	R	1	IN	AVERAGE DEPTH OF SNOW ON GROUND
SDRSP	R	1	-	SMALLEST VALUE OF DRSP
SDSC	R	1	-	SECOND DIFFERENTIAL OF SINE CURVE MAGNITUDE
SE	R	1	IN	CURRENT DAILY SNOW EVAPORATION
SERA	R	22	CFS	ACCUMULATED ABSOLUTE DIFFERENCES BETWEEN RECORDED AND SYNTHESIZED DAILY STREAMFLOWS FOR INTERVAL
SERAV	R	1	CFS	AVERAGE INTERVAL ABSOLUTE DIFFERENCE BETWEEN RECORDED AND SYNTHESIZED DAILY STREAMFLOWS
SERR	R	22	CFS	ACCUMULATED DIFFERENCES BETWEEN RECORDED AND SYNTHESIZED DAILY STREAMFLOWS FOR INTERVAL
SERRV	R	1	CFS	AVERAGE INTERVAL DIFFERENCE BETWEEN RECORDED AND SYNTHESIZED DAILY STREAMFLOWS
SESF	R	22	CFS	STANDARD ERROR OF SYNTHESIZED FLOWS BY MAGNITUDE INTERVAL
SET	R	1	IN	CURRENT HOURLY SOIL EVAPOTRANSPIRATION
SFDX	R	1	-	SUMMER FLOW DEVIATION INDEX
SFMD	R	1	-	SNOW FROZEN MOISTURE DENSITY
SGMD	I	1	-	STORAGE GAGE MOVING DAY (WHEN IT IS MOVED DURING WATER YEAR)
SGRT	I	1	-	STORAGE GAGE READING TIME
SGRT2	I	1	-	SECOND STORAGE GAGE READING TIME
SHM	R	1	-	SYNTHESIZED HYDROGRAPH MULTIPLIER
SHPF	R	1	CFS	SYNTHESIZED HYDROGRAPH PEAK FLOW
SIAC	R	1	-	SEASONAL INFILTRATION ADJUSTMENT CONSTANT
SIAM	R	1	-	SEASONAL INFILTRATION ADJUSTMENT MULTIPLIER
SIF	R	1	CFS	SYNTHESIZED INTERFLOW
SIFRS	R	3,20	CFS	SYNTHESIZED INTERFLOW DURING THE FIRST THREE DAYS OF EACH RECESSION SEQUENCE
SINDEX	R	1		NO OF SIMULATED HOURLY STORM SAMPLES
SISRX	R	1	-	SMALL INCREMENTAL STORAGE ROUTING INDEX
SMFX	R	1	-	SYNTHESIZED MONTHLY FLOW INDEX
SNTRI	I	1	-	SAVED NUMBER OF TIME ROUTING INCREMENTS
SOFMD	R	1	-	SUM OF OVERLAND FLOW MONTH DEVIATIONS
SOFRF	R	1	-	SNOW OVERLAND FLOW ROUTING FACTOR
SOFRFI	R	1	-	SNOW OVERLAND FLOW ROUTING FACTOR IMPERVIOUS SURFACES
SPBF	R	1	IN	SYNTHESIZED PERIOD BASE FLOW
SPBFLW	R	1	-	SNOW PACK BASIC MAXIMUM FRACTION IN LIQUID WATER
SPDR	R	1	IN	SYNTHESIZED PERIOD DIRECT RUNOFF
SPIF	R	1	IN	SYNTHESIZED PERIOD INTERFLOW
SPLW	R	1	IN	SNOW PACK LIQUID WATER CONTENT
SPLWC	R	1	IN	SNOWPACK LIQUID WATER HOLDING CAPACITY
SPM	R	1	-	SNOW PRECIPITATION MULTIPLIER
SPOF	R	1	CFS	SYNTHESIZED PERIOD OVERLAND FLOW (INCLUDING CHANNEL PRECIPITATION)
SPTF	R	1	IN	SYNTHESIZED PERIOD TOTAL FLOW
SPTW	R	1	IN	SNOW PACK TOTAL WATER CONTENT
SPTWCC	R	1	IN	SNOWPACK MINIMUM TOTAL WATER FOR COMPLETE BASIN COVERAGE
SQ	R	400	CFS	SIMULATED HOURLY STORM SAMPLES
SQER	R	22	CFS	ACCUMULATED SQUARES OF DIFFERENCES BETWEEN RECORDED AND SYNTHESIZED DAILY STREAMFLOWS
SQPKD	R	1	-	SUM OF SQUARED PEAK DIFFERENCES
SRR	R	5,170	CFS	STORM RUNOFF ROUTED DOWN CHANNELS
SRX	R	1	-	CURRENT STORAGE ROUTING INDEX
SSERA	R	1	CFS	ACCUMULATED ABSOLUTE DIFFERENCES BETWEEN RECORDED AND SYNTHESIZED FLOWS OVER INTERVALS
SSERAV	R	1	CFS	OVERALL AVERAGE ABSOLUTE DIFFERENCE BETWEEN RECORDED AND SYNTHESIZED FLOWS
SSERR	R	1	CFS	ACCUMULATED DIFFERENCES BETWEEN RECORDED AND SYNTHESIZED FLOWS OVER INTERVALS
SSERRV	R	1	CFS	OVERALL AVERAGE DIFFERENCE BETWEEN RECORDED AND SYNTHESIZED FLOWS
SSESF	R	1	CFS	ACCUMULATED STANDARD ERROR OF SYNTHESIZED FLOW OVER INTERVALS
SSQM	R	1	-	SUM OF THE SQUARES OF THE MONTHLY FLOW DEVIATIONS
SSQPKD	R	1	-	SMALLEST VALUE OF SQPKD
SSR	R	5,170	CFS	SYNTHESIZED STORM RUNOFF (NOT CHANNEL ROUTED)
SSRT	R	1	-	SQUARE ROOT OF OVERLAND FLOW SURFACE SLOPE
SSSQM	R	1	-	CURRENT SMALLEST ESTIMATE OF SSQM
STMD	R	1	-	SNOW TOTAL MOISTURE DENSITY
STMRQS	R	15,3	CFS	MEAN DAILY SIMULATED STORM SAMPLES

SUBWF	R	1	-	SUBSURFACE WATER FLOW OUT OF THE BASIN
SUZC	R	1	-	SEASONAL UPPER ZONE STORAGE CAPACITY FACTOR
SWSMD	R	1	-	SUM OF WET SUMMER MONTH DEVIATIONS
TANSM	R	1	IN	TOTAL ACCUMULATED NEGATIVE SNOWMELT (SNOW CHILLING)
TBRC	R	1	-	TOTAL BASE FLOW RECESSION DAYS
TDFP12	R	1	-	TIME OF DAILY FLOOD PEAK, 12-HOUR CLOCK
TDFP24	R	1	-	TIME OF DAILY FLOOD PEAK, 24-HOUR CLOCK
TDSF	R	1	CFS	TOTAL DAILY STREAMFLOW
TEH	R	1	DEGF	TEMPERATURE ESTIMATED FOR HOUR
TFCFS	R	1	CFS	CURRENT TOTAL FLOW
TFMAX	R	1	CFS	MAXIMUM TOTAL FLOW DURING CURRENT DAY
TFMRT	R	1	CFS	TOTAL STREAMFLOW AT MAXIMUM STREAM ROUTING TIME
TFSRX	R	1	-	TRIAL VALUE OF FSRX
TFX	R	1	CFS	TOTAL STREAMFLOW INDEX
THGR	R	1	IN/HR	TOTAL HOURLY GROSS RUNOFF
THSF	R	24	CFS	TOTAL HOURLY STREAMFLOW
TIRD	R	1	-	TOTAL INTERFLOW RECESSION DAYS
TITLE	A	20	-	TITLE OF CURRENT STATION YEAR (STREAMGAGE LOCATION AND DATE)
TMBF	R	12	IN	TOTALS OF DAILY BASE FLOW
TMFSIL	R	12	IN	TOTALS OF DAILY FOREST SNOW INT5R35PT90N-LOSS
TMIF	R	12	IN	TOTALS OF DAILY INTERFLOW
TMNET	R	12	IN	TOTALS OF DAILY NET EVAPOTRANSPIRATION
TMOF	R	12	IN	TOTALS OF DAILY OVERLAND FLOW
TMPET	R	12	IN	TOTALS OF DAILY POTENTIAL EVAPOTRANSPORATION
TMPREC	R	12	IN	TOTALS OF DAILY PRECIPITATION
TMRPM	R	12	IN	TOTALS OF DAILY RAIN PLUS MELT
TMRTF	R	12	SFD	TOTALS OF DAILY RECORDED TOTAL FLOW
TMSE	R	12	IN	TOTALS OF DAILY STREAM EVAPORATION
TMSME	R	12	IN	TOTALS OF DAILY SNOW EVAPORATION
TMSTF	R	12	SFD	TOTALS OF DAILY SYNTHESIZED TOTAL FLOW
TMSTFI	R	12	IN	TOTALS OF DAILY SYNTHESIZED TOT L&OLOW&R5&RECHV
TMTFCY	R	12	SFD	TOTALS OF DAILY TOTAL FLOW BY CALENDAR YEAR
TMTFWY	R	12	SFD	TOTALS OF DAILY TOTAL FLOW BY WATER YEAR
TOFR	R	1	IN	CURRENT TOTAL OVERLAND FLOW RUNOFF
TPLR	R	1	-	TOTAL TO PERVIOUS LAND RATIO
TRHF	R	1	IN/HR	CURRENT TIME ROUTED HYDROGRAPH FLOW
TRHV	R	1	-	TOTAL RECORDED HYDROGRAPH VOLUME
TRIP	I	1	-	COUNTER SPECIFYING PROGRAM PORTIONS
TSHV	R	1	-	TOTAL SYNTHESIZED HYDROGRAPH VOLUME
TSRX	R	7	-	ARRAY OF TRIAL STORAGE ROUTING INDICES
T20OFH	R	21	IN	TOP 20 VALUES DURING THE YEAR OF HOURLY OVERLAND FLOW
T20PRH	R	21	IN	TOP 20 VALUES DURING THE YEAR OF HOURLY PRECIPITATION
UHFA	R	59	IN	UNROUTED HYDROGRAPH FLOW ARRAY
URHF	R	1	IN	CURRENT UNROUTED HYDROGRAPH FLOW
UZC	R	1	IN	UPPER ZONE STORAGE CAPACITY
UZINFX	R	1	-	UPPER ZONE INFILTRATION INDEX
UZINLZ	R	1	IN/HR	CURRENT UPPER ZONE INFILTRATION TO LOWER ZONE
UZRX	R	1	-	UPPER ZONE MOISTURE RETENTION INDEX
UZS	R	1	IN	CURRENT UPPER ZONE STORAGE
VDCY	R	366	-	VALUE DATED BY CALENDAR DAY
VDMD	R	12	-	VALUE DATED BY MONTH DAY
VINTCR	R	1	IN	VEGETATIVE INTERCEPTION - CURRENT RATE PER PERIOD
VINTMR	R	1	IN/HR	VEGETATIVE INTERCEPTION - MAXIMUM RATE
VWIN	R	1	SFD	VOLUME OF AN INCH OF RUNOFF FROM WATERSHED
WAPV	R	1	-	WEIGHTED AVERAGE PARAMETER VALUE
WCFS	R	1	CFS	WATERSHED CFS EQUALLING ONE INCH PER HOUR
WEIFS	R	1	IN	WATER ENTERING INTERFLOW STORAGE
WFDX	R	1	-	WINTER FLOW DEVIATION INDEX
WI	R	1	IN	WATER INFILTRATION
WSBIT	R	1	-	WATERSHED BIT FOR RESTRUCTURING TIME-AREA HISTOGRAM
WSG	R	1	-	WEIGHTING FACTOR FOR STORAGE RAIN GAGE
WSG2	R	1	-	SECOND WEIGHTING FACTOR FOR STORAGE RAIN GAGE
WSM	R	1	-	NUMBER OF WET SUMMER MONTHS
WT4AM	R	1	DEGF	AVERAGE 4 A.M. TEMPERATURE OVER WATERSHED
WT4PM	R	1	DEGF	AVERAGE 4 P.M. TEMPERATURE OVER WATERSHED
WYEAR	I	1		ENDING OF WATER YEAR
XDNFS	R	1	-	INDEX DENSITY OF NEW-FALLEN SNOW
XELR	R	1	-	RAIN INDEX FOR ESTIMATING LAPSE RATE 0.0 = DRY, 4.0 = RAIN
XMPFT	R	12	-	INDEX OF MONTHLY PREDOMINATE FLOW TYPE
YEAR	I	1	-	LAST TWO DIGITS OF CURRENT YEAR
YR1	I	1	-	LAST TWO DIGITS OF FIRST CALENDAR YEAR IN WATER YEAR
YR2	I	1	-	LAST TWO DIGITS OF SECOND CALENDAR YEAR IN WATER YEAR
YTITLE	A	20	-	YEAR TITLE

ORIGINAL PAGE IS
OF POOR QUALITY

APPENDIX B

SOURCE PROGRAM LISTINGS

TABLE OF CONTENTS

APPENDIX B

SOURCE PROGRAM LISTINGS

```
C         MAIN ********  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 197
C         BASED ON STANFORD WATERSHED WATERSHED MODELS III & IV
C
C     ISFLAG IS SET TO BRANCH AS FOLLOWS:
C          1 - BRANCH TO NOMINAL KWM INITIAL RUN
C          2 - PLOTS ONLY, NO NEXT YEAR OF DATA
C          3 - NO PLOTS, NEXT YEAR DATA WITH NEW WATERSHED PARAMETERS
C          4 - NO PLOTS, NEXT YEAR DATA WITH OUT NEW WATER SHED PARAMETERS
C          5 - PLOTS , NEXT YEAR DATA WITH NEW WATERSHED PARAMETERS
C          6 - PLOTS, NEXT YEAR DATA WITHOUT NEW WATERSHED PARAMETERS
C        7 - PLOTS, WITH SIM AND OBS SUPERIMPOSED ONLY(BOTH MONTH AND YER
C        8 - PLOTS, WITH SIM AND OBS SUPERIMPOSED ONLY(MONTHS SCALE)
C        9 - NO PLOTS, NO NEXT YEAR DATA
C
C
      DIMENSION  TITLE(18), DRSF(366), DSSF(366), CONOPT(16),
     1 THSFD(744,3),PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6),Q(337),
     A TMRTF(12),TMSTF(15,3),
     A  CTRI(99),FIRR(15),RICY(37),DPSE(366),BTRI(99)
      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
     2KFLAG,ICFLAG,
     1IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
     2  NCTRI,CTRI,FIRR,RICY,DPSE,BDDFSM,SP3FLW,SPTWCC,SPM,ELDIF,
     3  XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
     A SATRI(99),UHFA(99),
     BMNRD,
     4  FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
     5  BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,
     6  BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
     7  IFRL,LSHFT,NBTRI,FNTRI,MXTFI,NCSTRI,BTFI,TFCFS,FPAET,FPER,
     8  TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPOR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
     9  RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
     A  WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
     B  SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
     XDEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
     YTOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
     ZTSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
     WTORARY(1,6,1),
     CDRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)
      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
     1TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
     2EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
     3TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,
     4TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
     5TRHF,
     5SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
      DIMENSION RPLOTC(1832),RCOMMA(12087)
      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))
      DIMENSION CRFMI(22),DDIW(366),
     1  DMNT(366), DMXT(366),DRGPM(366), DRHP(366,24),
     2  DRSGP(366), DPET(366),EDLZS(366),
     3 EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
     4 EMUZC(15,3),EMUZS(15,3),EPCM(12),
     5 SERA(22), SERR(22), SESF(22), SQER(22),
     6 THSF(24),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
     7 TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMSE(15,3),
     8 TMSNE(12),TMSTFI(15,3),T200FH(21),T20PRH(21),
     9 TMRTFI(12),JULDAT(6),TFMAXY(366)
      REAL LZS,IFS
      REAL MNRD
      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY
      INTEGER PEAKS,PHRS,CONOPT,NSPTS
      INTEGER SINDEX,SINDET,MPDAY(15)
      INTEGER NUNIT/2/,IER
      INTEGER ICCN/1/
      REAL SUM(3),APREC(3),QOUT(15)
            INTEGER MONTHS(64),PMONTH,PDATE,EMONTH,EDATE,WYEAR,DATES,PDAY
      DATA MONTHS/'JAN ',0,0,0,4,'FEB ',31,31,31,5,'MAR ',59,59,59,6,
     1'APR ',90,90,90,7,'MAY ',120,120,120,8,'JUN ',151,151,151,9,
     2'JUL ',181,181,181,10,'AUG ',212,212,212,11,'SEPT',243,243,243,12,
     3'OCT ',273,273,274,1,'NOV ',304,304,305,2,'DEC ',334,334,335,3,
     4'JAN ',365,365,366/
      READ (5,9)  TITLE
    9 FORMAT(18A4)
      CALL  DSUPDT
      GO TO 806
  805 CONTINUE
      BACKSPACE 5
      BACKSPACE 5
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      REWIND 11
      REWIND 18
      CALL TAPCON(NUNIT,ICON,IER)
      CALL DSUPDT
806   CONTINUE
       ISFLAG = 1
      JPLOT=0
      IDFLAG=0
      IENDFG=0
      ID=1
      ID=5
      WRITE (6,9990)   IDFLAG,IENDFG,ISFLAG,ID
      CALL READ(IRSTRT)
      CALL READ(PMONTH,PDATE,EMONTH,EDATE,NYEAR,WYEAR)
      DO 101  KRD=1,16
101   CALL READ(CONOPT(KRD))
      PMONTH=((PMONTH-1)*5)+1
      MSBDIC=MONTHS(PMONTH)
      LEAPY=PMONTH+1
      IF  (MOD(WYEAR,4).EQ.0)   LEAPY=PMONTH+2
      IF  (MOD(NYEAR,4).EQ.0)   LEAPY=PMONTH+3
      DATES=MONTHS(LEAPY)+PDATE
      PDAY=(MONTHS(LEAPY))
      MONTH=MONTHS(PMONTH+4)
      EMONTH=((EMONTH-1)*5)+1
      LDAY=MONTHS(LEAPY+5)
      IF  (PMONTH.EQ.EMONTH)  EDATE=MONTHS(LEAPY)+EDATE
      IF((PMONTH.NE.EMONTH).AND.(MONTHS(EMONTH+4).NE.4))
     1EDATE=(MONTHS(LEAPY+5)+EDATE)
      IF((PMONTH.NE.EMONTH).AND.(MONTHS(EMONTH+4).EQ.4))
     1EDATE=MONTHS(2)+EDATE
      IF  ((MONTH.EQ.5).AND.(EDATE.EQ.60))  EDATE=366
      ISTART=CONOPT(15)
      NFP=3
      IF  (CONOPT(15).EQ.2)   NFP=1
      CALL READ(DT,AREAI,NQ,LOOKUP,IPLOT)
      ID=6
      DO 503 I=1,NQ
      CALL READ (Q(I))
503   CONTINUE
      WRITE(6,9990)          IDFLAG,IENDFG,ISFLAG,ID
 9990 FORMAT(/3X,'      IDFLAG=   ',I5,
     *'      IENDFG=   ',I5,'      ISFLAG=   ',I5,'      MAIN=   ',I5)
20    GO TO (1,2,3),ISTART
      ISFLAG=0
      ID=2
      WRITE(6,9990)          IDFLAG,IENDFG,ISFLAG,ID
       CALL KWMAIN(ISFLAG)
      CALL  WORSTC(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
      CALL   ZEROP(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
      CALL  FORCST(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)
      CALL  INT(NQ,Q,DT,AREAI,LOOKUP,IPLOT,NFP,MSBDIC,MPDAY,DIFFS,
     1DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,DIFFPR,DIFPP,QOUT,
     2QMAX,PHRO)
      CALL OUTPUT(DIFFS,DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,
     1DIFFPR,DIFFPP,QOUT,QMAX,PHRO,MSBDIC,MPDAY)
      IF  (IRSTRT.EQ.1)  GO TO 805
      STOP
16    CONTINUE
    1 CONTINUE
      ID=6
      WRITE(6,9990)          IDFLAG,IENDFG,ISFLAG,ID
      CALL  WORSTC(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
      CALL   ZEROP(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
      CALL  FORCST(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)
      CALL  INT(NQ,Q,DT,AREAI,LOOKUP,IPLOT,NFP,MSBDIC,MPDAY,DIFFS,
     1DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,DIFFPR,DIFPP,QOUT,
     2QMAX,PHRO)
      CALL OUTPUT(DIFFS,DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,
     1DIFFPR,DIFFPP,QOUT,QMAX,PHRO,MSBDIC,MPDAY)
      IF  (IRSTRT.EQ.1)  GO TO 805
      STOP
17    CONTINUE
      GO TO 20
2     CONTINUE
      CALL  PASTRN(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)
      CALL  INT(NQ,Q,DT,AREAI,LOOKUP,IPLOT,NFP,MSBDIC,MPDAY,DIFFS,
     1DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,DIFFPR,DIFPP,QOUT,
     2QMAX,PHRO)
      CALL OUTPUT(DIFFS,DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,
     1DIFFPR,DIFFPP,QOUT,QMAX,PHRO,MSBDIC,MPDAY)
      IF  (IRSTRT.EQ.1)  GO TO 805
      STOP
3     CONTINUE
      READ  (11)   (RPLOTC(I),I=1,1832)
      READ  (18)    (RCOMMA(I),I=1,12087)
      CALL  READ  (GWS,UZS,LZS,BFNX,IFS,UZC,SIAM)
      REWIND 11
      REWIND 18
      WRITE  (11)    (RPLOTC(I),I=1,1832)
      WRITE  (18)    (RCOMMA(I),I=1,12087)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE KWMAIN(IBFLAG)
       COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMRUS(121,6),DPY,TITLE,
     2KFLAG,IDFLAG,
     1IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
     2  NCTRI,CTRI,FIRR,RICY,DPSL,BDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,
     3  XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,FMPF,RGPMB,AREA,FIMP,
     A SATRI,UHFA,
     BMNRD,
     4  FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
     5  BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSPX,CHCAP,EXQPV,
     6  BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
     7  IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAET,FPFR,
     8  TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPDR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
     9  RHFD,URHF,AMIF,AMNET,AMPET,AMSNF,AMFSIL,SASFX,SARAX,SRX,VWIN,
     A  WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
     B  SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
     XDEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
     YTOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(9,1),
     ZTSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
     WTORARY(1,6,1),
     CDRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)
       COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,IMBF,
     1TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
     2EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
     3TMSTFI,T2COFH,T20PRH,TMRTFI,JULDAT,
     4TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
     5TRHF,
     5SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
C  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)
C     BASED ON STANFORD WATERSHED MODELS III & IV
       DIMENSION BTRI(99),CONOPT(16),CRFMI(22),CTRI(99),DDIW(366),
     1  DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
     2  DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
     3 EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
     4 EMUZC(15,3),EMUZS(15,3),EPCM(12),FIPR(15),MEDCY(12),MEDWY(12),
     5  RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),
     6 THSF(24),TITLE(13),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
     7 TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
     8 TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T20OFH(21),T20PRH(21),
     9 UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
     A PEAKS(6),PHRS(6),NSPTS(6),THSFDM(5)
       DIMENSION RPLOTC(1532),RCOMMA(12037)
       EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))
       LOGICAL  LSHFT
       INTEGER  CDSDR,CN,CONOPT,DATE,DAY,DPY,FHSGD,HOUR,HRF,HRL,PDAY,
     1  PRD,RHPD,RHPH,RSRD,SGMD,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX
       INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY
       REAL  IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT
       DATA  MEDCY/  0,         31,59,90,120,151,181,212,243,273,304,334/
       DATA  MEDWY/304,334,365,31,59,90,120,151,181,212,243,273        /
       REAL     MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD
       REAL  SSQM,SSQDI,SSQMI,VDRSF,VDSSF
       REAL  VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
     *        SMSQM
       REAL  MDRSF,MDSSF,MMRSF,MMSSF
   98 IF (IBFLAG) 99,100,116
   99 IBFLAG = 13
      RETURN
  100 CONTINUE
      CALL READ(GWS,UZS,LZS,BFNX,IFS,UZC,SIAM)
      DO 102 KIA = 1,99
      SATRI(KIA) = 0.0
      CTRI(KIA) = 0.0
      BTRI(KIA) = 0.0
  102 UHFA(KIA) = 0.0
      CALL READ(NCTRI)
      DO 103 KRD = 1, NCTRI
  103 CALL READ(CTRI(KRD))
      IF(CONOPT(7) .NE. 1) GO TO 110
      DO 104 KRD = 1,15
  104 CALL READ(FIRR(KRD))
      DO 105 KRD = 1, 37
  105 CALL READ(RICY(KRD))
      DO 106 KRD = 274,360,10
  106 CALL READ(DPSE(KRD))
      DO 107 KRD = 1,273,10
  107 CALL READ(DPSE(KRD))
      DO 109 IDAY2 = 1, 9
      DO 108 IDAY1 = 274,360,10
      DAY = IDAY1 + IDAY2
  108 DPSE(DAY) = DPSE(IDAY1)
      DO 109 IDAY1 = 1,273,10
      DAY = IDAY1 + IDAY2
      IF(DAY .GT. 273) GO TO 109
      DPSE(DAY) = DPSE(IDAY1)
  109 CONTINUE
      DPSE(366) = DPSE(59)
      DPSE(365) = DPSE(363)
      DPSE(364) = DPSE(363)
      CALL READ(BDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,XDNFS,FFOR,FFSI,MRNSM,
     1  DSMGH,PXCSA)
```

```
110 CALL READ(RMPF)
    CALL READ (RGPMR,AREA,FIMP,FWTR)
    CALL READ (VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,BIVF)
    CALL READ (OFSS,OFSL,OFMN,OFMNIS,IFRC)
    CALL READ (CSRX,FSRX,CHCAP,EXOPV,BFNLR,BFRC)
    BFHRC = BFRC**(1.0/24.0)
    BFRL = -ALOG(BFHRC)
    BFNRL = 0.0
    IF(BFNLR .LT. 0.00001 .OR. BFNLR .GT. 0.9999) GO TO 111
    BFNHR = BFNLR**(1.0/24.0)
    BFNRL = -ALOG(BFNHR)
111 IFPRC = IFRC**(1.0/96.0)
    IFRL = -ALOG(IFPRC)
    LSHFT = .FALSE.
    IF(CONOPT(13) .NE. 1) GO TO 113
    NBTRI = NCTRI
    FNTRI = NCTRI
    MXTRI = (10.0**EXQPV)*FNTRI + 0.5
    IF(MXTRI .GE. 98) WRITE(6,1)
  1 FORMAT(29HWARNING: EXQPV ARRAY OVER RUN)
    NCSTRI = 99
    DO 112 KIA = 1, NBTRI
112 BTRI(KIA) = CTRI(KIA)
    TFCFS = 1.0
    CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
   1  TFCFS)
113 EPAET = 0.0
    FPER = 1.0 - FIMP - FWTR
    IF(FPER .GT. 0.01) GO TO 114
    TPLR = 100.0
    FPER = 0.01
    GO TO 115
114 TPLR = (1.0 - FWTR)/FPER
115 VINTCR = 0.25*VINTMR
    HSE = 0.0
    NRTRI = 0
    SPIF = 0.0
    CBF = GWS*BFRL*(1.0 + BFNRL*BFNX)
    SPDR = 0.0
    OFUS = 0.0
    OFUSIS = 0.0
    OFR = 0.0
    OFRIS = 0.0
    PEIS = 0.0
    RHFO = 0.0
    URHF = 0.0
    AMIF = 0.0
    AMNET = 0.0
    AMPET = 0.0
    AMSNE = 0.0
    AMFSIL = 0.0
    SASFX = 0.0
    SARAX = 0.0
    SRX = CSRX
    VWIN = 26.8888*AREA
    WCFS = 24.0*VWIN
    RHFMC = 0.025/WCFS
    TFCFS = CBF*WCFS
    SSRT = SQRT(OFSS)
    OFRF = 1020.0*SSRT/(OFMN*OFSL)
    OFRFIS = 1020.0*SSRT/(OFMNIS*OFSL)
    EQDF = 0.00982*((OFMN*OFSL/SSRT)**0.6)
    EQDFIS = 0.00982*((OFMNIS*OFSL/SSRT)**0.6)
    SOFRF = OFRF
    SOFRFI = OFRFIS
    SDEPTH = 0.0
    IYR=0
    MI=0
    NI=0
    MULT=0
    MULTI=CONOPT(10)
    ID=1
    WRITE(6,9990)  MULTI,  CONOPT(10),  CONOPT(15),  IBFLAG, ID
9990 FORMAT(/3X,'    MULTI=  ',I5,'    CONOPT(10)=  ',I5,
   *'    CONOPT(15)=  ',I5,'    IBFLAG=  ',I5,'    KWMAIN=  ',I5)
    ASM = 0.0
    IF(CONOPT(7) .EQ. 0) GO TO 116
    WT4AM = 60.0
    WT4PM = 60.0
    SAX = 15.0
    TANSM = 0.0
    SPTW = 0.0
    STMD = 0.7
    SFMD = 0.7
    ASMRG = 0.0
116   CONTINUE
C  BEGIN NEW YEAR
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
  117 BYLZS = LZS
      BYUZS = UZS
      BYGWS = GWS
      BYIFS = IFS
      DO 118 KIA = 1,22
      CRFMI(KIA) = 0.0
      SESF(KIA) = 0.0
      SERR(KIA) = 0.0
      SERA(KIA) = 0.0
  118 SQCR(KIA) = 0.0
      RGPM = RGPMB
      DO 119 KIA = 1,21
      T20OFH(KIA) = 0.0
  119 T20PRH(KIA) = 0.0
      DO 120 KIA = 1,12
  120 EPCM(KIA) = 1.0
      RDPT = 0.0
      PDAY = 274
903   CALL READ(YR1,YR2)
      TODARY(1,IYR)=YR2
      TOMARY(1,IYR)=YR2
      DPY = 365
      IF(MOD(YR2,4) .EQ. 0) DPY = 366
      IF(CONOPT(1) .EQ. 1) CALL READ(CDSDR,NDSDR)
      NDSDP = 0
      MEDWY(5) = 59
      IF(DPY .EQ. 366) MEDWY(5) = 366
C   READ EVAPORATION DATA
      IF(CONOPT( 3) .NE. 1) GO TO 125
      DO 121 KRD = 274,360,10
  121 CALL READ(DPET(KRD))
      DO 122 KRD = 1,273,10
  122 CALL READ(DPET(KRD))
      DO 124 IDAY2 = 1,9
      DO 123 IDAY1 = 274,360,10
      DAY = IDAY1 + IDAY2
  123 DPET(DAY) = DPET(IDAY1)
      DO 124 IDAY1 = 1,273,10
      DAY = IDAY1 + IDAY2
      IF(DAY .GT. 273) GO TO 124
      DPET(DAY) = DPET(IDAY1)
  124 CONTINUE
      DPET(366) = DPET(59)
      DPET(365) = DPET(363)
      DPET(364) = DPET(363)
      GO TO 127
  125 IF(CONOPT( 3) .EQ. 2) GO TO 130
      DAY = 274
  126 CALL READ (DPET(DAY))
      IF(DAY .EQ. 273) GO TO 127
      CALL DAYNXT(DAY, DPY)
      GO TO 126
  127 DO 128 MONTH = 1,12
  128 CALL READ(EPCM(MONTH))
      IF(EPAET .NE. 0.0) GO TO 133
      DO 129 DAY = 1,DPY
  129 EPAET = EPAET + DPET(DAY)
      IF(EPCM(6) .NE. 1.0) EPAET = 0.7*EPAET
      GO TO 131
  130 CALL READ(EPAET,MNRD)
      EMAET = EPAET*(365.0 + MNRD)/404.0
      CALL EVPDAY(DPET,EMAET)
  131 AETX = 24.0*EPAET/365.0
      AEX96 = 1.2*AETX
      AEX90 = 0.3*AETX
      SIAM = 1.2**SIAC
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZS/LZC)
      IF(UZC .LT. 0.25) UZC = 0.25
      SGRT = 0
      DO 132 DAY = 1,366
      DDIW(DAY) = 0.0
      DRSF(DAY) = 0.0
      DRGPM(DAY) = RGPMB
      DRSGP(DAY) = 0.0
      DO 132 HOUR = 1,24
  132 DRHP(DAY,HOUR) = 0.0
  133 IF(CONOPT(9) .NE. 1) GO TO 135
      DAY = 274
      DRSF(366) = 0.0
  134 CALL READ(DRSF(DAY))
      CALL DAYNXT(DAY, DPY)
      IF(DAY .NE. 274) GO TO 134
  135 IF(CONOPT(11) .NE. 1) GO TO 137
      DAY = 274
      DDIW(366) = 0.0
  136 CALL READ(DDIW(DAY))
      CALL DAYNXT(DAY, DPY)
      IF(DAY .NE. 274) GO TO 136
```

```
  137 IF(CONOPT(7) .EQ. 0) GO TO 139
      DAY = 274
  138 CALL READ(DMXT(DAY), DMNT(DAY))
      CALL DAYNXT(DAY, DPY)
      IF(DAY .NE. 274) GO TO 138
  139 CALL READ(NSGRD)
      IF(NSGRD .EQ. 0) GO TO 141
      CALL READ(WSG,SGRT)
      IF(CONOPT( 8) .EQ. 1) CALL READ(WSG2,SGRT2,SGMD)
      DO 140 KRD = 1,NSGRD
      CALL READ(ISGRD)
  140 CALL READ(DRSGP(ISGRD))
C   READ RECORDING RAIN GAGE HOURLY TOTALS
  141 CALL READ(IWBG,YEAR,MONTH,DATE,CN)
C   PUNCH NO NUMBER AFTER CN ON YEAR .EQ. 98 CARD
      IF(YEAR .GE. 98) GO TO 144
      HRF = 12*(CN - 1) + 1
      HRL = 12*(CN - 1) + 12
      DAY = MEDCY(MONTH) + DATE
      DO 142 HOUR = HRF, HRL
  142 CALL READ(DRHP(DAY,HOUR))
      IF(DPY .NE. 366 .OR. MONTH .NE. 2 .OR. DATE .NE. 29) GO TO 141
      DO 143 HOUR = HRF, HRL
      DRHP(366,HOUR) = DRHP(60,HOUR)
  143 DRHP(60,HOUR) = 0.0
      GO TO 141
C   CALCULATE PRECIPITATION WEIGHTING FACTORS
  144 DAY = 274
      IF(NSGRD .EQ. 0) GO TO 151
      PDAY = 274
      RDPT = 0.0
  145 EHSGD = SGRT
      IF(SGRT .EQ. 0) EHSGD = 24
      EHSGDF = EHSGD
  146 CONTINUE
      DO 150 HOUR = 1,24
      RDPT = RDPT + DRHP(DAY,HOUR)
      IF(HOUR .NE. EHSGD) GO TO 150
      IF(RDPT .LE. 0.0) GO TO 147
      IF(SGRT .EQ. 0) PDAY = DAY
      DRGPM (PDAY) = (DRSGP(DAY)*WSG + RDPT*(1.0 - WSG))/RDPT
      IF(CONOPT(3) .NE. 0) DPET(PDAY) = 0.5*DPET(PDAY)
      IF(SGRT .NE. 0) PDAY = DAY
      RDPT = 0.0
      GO TO 150
  147 IF(DRSGP(DAY) .LE. 0.0) GO TO 149
      DO 148 KHOUR = 1,EHSGD
  148 DRHP(DAY,KHOUR) = (WSG*DRSGP(DAY))/EHSGDF
  149 IF(SGRT .NE. 0) PDAY = DAY
  150 CONTINUE
      CALL DAYNXT(DAY,DPY)
      IF(DAY .EQ. 274) GO TO 151
      IF(CONOPT(8) .EQ. 0) GO TO 146
      IF(DAY .NE. SGMD) GO TO 146
      WSG = WSG2
      SGRT = SGRT2
      GO TO 145
151   BACKSPACE 11
      WRITE (11) (RPLOTC(I),I=1,1832)
      BACKSPACE 18
      WRITE (18) (RCOMMA(I),I=1,12087)
      RETURN
      END
```

```
      SUBROUTINE WORSTC(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMPDS(121,6),DPY,TITLE,
     2KFLAG,IDFLAG,
     1IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
     2  NCTRI,CTRI,FIRR,RICY,DPSE,ADDFSM,SPMFLW,SPTWCC,SPM,ELDIF,
     3  XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
     A SATRI,UHFA,
     BMNRD,
     4  FWTR,VINTMR,BUZC,SUZC,LZC,FTLF,SUBWF,GWETF,SIAC,BMIR,
     5  BIVF,CFSS,OFSL,OFMN,OFMNIS,IFRC,CSPX,CSRX,CHCAP,EXQPV,
     6  BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHPC,BFRL,BFNRL,BFNHF,IFPRC,
     7  IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAFT,FPCR,
     8  TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPDR,CHUS,UHUSIS,OFR,OFRIS,PEIS,
     9  RHFO,URHF,AMIF,AMNET,AMPFT,AMSNE,AMFSIL,SASFX,SARAX,SRX,VAIN,
     A  WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
     B  SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
     XDEPEND(2),VARIN(2),JPTS,JULPI,IYR,TODARY(5,1),
     YTOMARY(7,1),TOSARY(5,6,1),TSCARY(6,1),TSMARY(8,1),
     ZTSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
     WTORARY(1,6,1),
     CDRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)
      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
     1TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
     2EPCM,SERA,SERR,SESF,SOER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
     3TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,
     4TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
     5TRHF,
     5SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
C   KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1979)
C     BASED ON STANFORD WATERSHED MODELS III & IV
      DIMENSION BTRI(99),CONOPT(16),CRFMI(22),CTRI(99),DDIW(366),
     1  DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
     2  DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
     3 EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
     4 EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
     5  RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SOER(22),
     6 THSF(24),TITLE(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
     7 TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
     8 TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T200FH(21),T20PRH(21),
     9 UHFA(99),TMRTF1(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
     A PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)
      LOGICAL  LSHFT
      INTEGER  CDSCR,CN,CONOPT,DATE,DAY,DPY,EHSGD,HOUR,HPF,HRL,PDAY,
     1  PRD,RHPD,RHPH,RSHD,SGMD,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX
      INTEGER TOMARY,TSMARY,TODARY,TSCARY,TOSARY,TSSARY
      INTEGER  DATES,EDATE
      DIMENSION RPLOTC(1832),RCOMMA(12087)
      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))
      REAL   IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT
      DATA   MEDCY/  0,          31,59,90,120,151,181,212,243,273,304,334/
      DATA   MEDWY/304,334,365,31,59,90,120,151,181,212,243,273         /
      REAL       MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD
      REAL   SSQM,SSQDI,SSQMI,VDRSF,VDSSF
      REAL   VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
     *         SMSQM
      REAL   MDRSF,MDSSF,MMRSF,MMSSF
      BACKSPACE 11
      READ   (11) (RPLOTC(I),I=1,1832)
      BACKSPACE 18
      READ(18) (RCOMMA(I),I=1,12087)
 150  CONTINUE
      DATE=PDATE
      DAY=DATES
      MDAY=PDAY
      SINDEX=0
      MAXI=0
      AMRPM = 0.0
      AMPREC = 0.0
      AMBF = 0.0
      AMSE = 0.0
      AMSTF = 0.0
      AMRTF = 0.0
      IF(DPY .EQ. 366) MEDWY(5)=366
      WRITE(6,3) (TITLE(KTA),KTA=1,18)
 3    FORMAT (1H1,10X,18A4,//)
      WRITE(6,4) MSBDIC
 4    FORMAT(1X,'WORSE CASE HOURLY CFS VALUES',//,1X,4A4)
C   BEGIN DAY LOOP
      IMONTH=MONTH
  152 TDSF = 0.0
      IF (MONTH.NE.4) GO TO 148
      IF (MDAY.NE. 31) GO TO 149
 148  IF (DATE.GT.(MOD(DAY,MDAY))) IMONTH=MONTH+1
 149  CONTINUE
      PET=EPCM(IMONTH)*DPET(DAY)
      PETU = PET
      TFMAX = 0.0
C   EVAPOTRANSPIRATION ADJUSTMENTS
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IF(CONOPT(7) .NE. 1) GO TO 153
      IF(DMXT(DAY) - 4.0*ELDIF .LT. 40.0) PET = 0.0
      IF(SPTW .GT. SPTWCC) PET = FFOR*PET
C  CALCULATION OF SNOW EVAPORATION
      IF(DMNT(DAY) .GT. 32.0 .OR. SPTW .LE. DPSE(DAY)) GO TO 153
      SE = DPSE(DAY)
      AMSNE = AMSNE + SE
      SPTW = SPTW - SE
      IF(SFMD .GT. 0.0) SDEPTH = SDEPTH - SE/SFMD
  153 DO 202 HOUR = 1,24
      IF((NSGRD .EQ. 0) .AND. (DRHP(DAY,HOUR) .NE. 0.0)  .AND. (PET .EQ.
     1  PETO) .AND. (CONOPT(3) .EQ. 1)) PET = 0.5*PET
  154 IF(HOUR .EQ. SGRT + 1) RGPM = DRGPM(DAY)
      IF(HOUR .EQ. 9) HSE = (FWTR*PET)/12.0
      IF(HOUR .EQ.21) HSE = 0.0
      PRH=1.2*DRHP(DAY,HOUR)
      AMPREC = AMPREC  + PRH
C  ENTER SNOWMELT SUBROUTINE
      IF(CONOPT(7) .EQ. 1) CALL SNOMEL(BDDFSM,SPTWCC,SPM,ELDIF,DAY,
     1  SPBFLW, XDNFS,FFOR,FFSI,HRNSM,DSMGH,SDEPTH,STMD, PXCSA,HOUR,
     2  SAX,SOFRF,OFRFIS,SOFRFI,AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,
     3  WT4AM,WT4PM,ASM,ASMRG,SASFX,SARAX,DMXT,DMNT,RICY,FIRR)
  155 AMRPM = AMRPM + PRH
  156 TOFR = 0.0
      ARHF = 0.0
C  15 MINUTE ACCOUNTING AND ROUTING LOOP
      DO 187 PRD = 1,4
      PEBI = 0.0
      PPI = 0.0
      OFR = 0.0
      OFRIS = 0.0
      WI = 0.0
      WEIFS = 0.0
      PMEUZS = 0.0
      PMELZS = 0.0
      PMEIFS = 0.0
      PMEOFS = 0.0
      PEP = 0.25*PRH
      IF(CONOPT(2) .EQ. 1) CALL PREPRD(RGPM,DRHP,DAY,HOUR,DPY,PRD,PEP,
     1  PRH)
      IF(PEP .GT. 0.0) GO TO 157
      IF(OFUS .GT. 0.0) GO TO 159
      IF(IFS .GT. 0.0) GO TO 170
      IF(NRTRI .GT. 0) GO TO 172
      TRHF = 0.0
      IF(RHFO .GT. 0.0) GO TO 181
      GO TO 184
C  RAINFALL UPPER ZONE INTERACTION
  157 IF(PEP .GE. VINTCR) GO TO 158
      UZS = UZS + PEP*TPLR
      VINTCR = VINTCR - PEP
      PPI = 0.0
      PEBI = 0.0
      PMEUZS = PEP
      IF(OFUS .GT. 0.0) GO TO 159
      GO TO 170
  158 PPI = PEP - VINTCR
      UZS = UZS + VINTCR*TPLR
      VINTCR = 0.0
      LZSR = LZS/LZC
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZSR)
      IF(UZC .LT. 0.25) UZC = 0.25
      UZRX = 2.0*ABS(UZS/UZC - 1.0) + 1.0
      FMR = (1.0/(1.0 + UZRX))**UZRX
      IF(UZS .GT. UZC) FMR = 1.0 - FMR
      PEBI = PPI*FMR
      PMEUZS = PEP - PEBI
      UZS = UZS + PPI - PEBI
C  LOWER ZONE AND GROUNDWATER INFILTRATION
  159 LZSR = LZS/LZC
      EID = 4.0*LZSR
      IF(LZSR .LE. 1.0) GO TO 160
      EID = 4.0 + 2.0*(LZSR - 1.0)
      IF(LZSR .LE. 2.0) GO TO 160
      EID = 6.0
  160 PEBI = PEBI + OFUS
      CMIR = 0.25*SIAM*BMIR/(2.0**EID)
      CIVM = BIVF*2.0**LZSR
      IF(CIVM .LT. 1.0) CIVM = 1.0
      PEAI = PEBI*PEBI/(2.0*CMIR*CIVM)
      WI = PEBI*PEBI/(2.0*CMIR)
      IF(PEBI .GE. CMIR) WI = PEBI - 0.5*CMIR
      IF(PEBI .GE. CMIR*CIVM) PEAI = PEBI - 0.5*CMIR*CIVM
      WEIFS = WI - PEAI
      IF(PEBI .LE. OFUS) GO TO 161
      PMELZS = (PEBI - WI)*((PEBI - OFUS)/PEBI)
      PMEIFS = WEIFS*((PEBI - OFUS)/PEBI)
      PMEOFS = PEAI*((PEBI - OFUS)/PEBI)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
  161 CONTINUE
      IF((PEAI - OFUS) .GT. 0.0) GO TO 162
      EQD = (OFUS + PEAI)/2.0
      GO TO 163
  162 EQD = EQDF*((PEAI - OFUS)**0.6)
  163 IF((OFUS + PEAI) .GT. (2.0*EQD)) EQD = 0.5*(OFUS + PEAI)
      IF((OFUS + PEAI) .LE. 0.001) GO TO 164
      OFR = 0.25*OFRF*((((OFUS + PEAI)*0.5)**1.67)*((1.0 + 0.6*((OFUS +
     1  PEAI)/(2.0*EQD))**3.0)**1.67)
      IF(OFR .GT. (0.75*PEAI)) OFR = 0.75*PEAI
  164 IF(FIMP .EQ. 0.0) GO TO 168
  165 PEIS = PPI + OFUSIS
      IF((PEIS - OFUSIS) .GT. 0.0) GO TO 166
      EQDIS = (OFUSIS + PEIS)/2.0
      GO TO 167
  166 EQDIS = EQDFIS*((PEIS - OFUSIS)**0.6)
  167 IF((OFUSIS + PEIS) .GT. (2.0*EQDIS)) EQDIS = 0.5*(OFUSIS + PEIS)
      IF((OFUSIS + PEIS) .LE. 0.01) GO TO 168
      OFRIS = 0.25*OFRFIS*((((OFUSIS + PEIS)*0.5)**1.67)*((1.0 + 0.6*((
     1  OFUSIS + PEIS)/(2.0*EQDFIS))**3.0)**1.67)
      IF(OFRIS .GT. PEIS) OFRIS = PEIS
  168 TOFR = TOFR + FPER*OFR + FIMP*OFRIS + PPI*FWTR
      OFUSIS = PEIS - OFRIS
      OFUS = PEAI - OFR
      IF(OFUS .GE. 0.001) GO TO 169
      LZS = LZS + OFUS
      OFUS = 0.0
      OFRIS = OFRIS + OFUSIS
      OFUSIS = 0.0
  169 LZRX = 1.5*ABS(LZS/LZC - 1.0) + 1.0
      FMR = (1.0/(1.0 + LZRX))**LZRX
      IF(LZS .LT. LZC) FMR = 1.0 - FMR*(LZS/LZC)
      PLZS = FMR*(PEBI - WI)
      PGW = (1.0 -FMR)*(PEBI - WI)*(1.0 - SUBWF)*FPER
      GWS = GWS + PGW
      BFNX = BFNX + PGW
      LZS = LZS + PLZS
      IFS = IFS + WEIFS*FPER
  170 SPIF = IFRL*IFS
      AMIF = AMIF + SPIF
      IFS = IFS - SPIF
      IF(IFS .GE. 0.0001) GO TO 171
      LZS = LZS + IFS
      IFS = 0.0
  171 UHFA(1) = FPER*OFR + PPI*FWTR + FIMP*OFRIS + SPIF
      SPOR = UHFA(1)
C  ROUTING
  172 IF(CONOPT(12) .NE. 1) GO TO 173
      URHF = URHF + 0.25*UHFA(1)
      IF(PRD .NE. 4) GO TO 181
      UHFA(1) = URHF
  173 TRHF = 0.0
      KTRI = NCTRI
      IF(CONOPT(13) .EQ. 1) KTRI = NCSTRI
  174 URHF = UHFA(KTRI)
      IF(URHF .LE. 0.0) GO TO 176
  175 TRHF = TRHF + URHF*CTRI(KTRI)
      IF(CONOPT(13) .EQ. 1 .AND. LSHFT .AND. KTRI .GE. 2) TRHF = TRHF +
     1  URHF*SATRI(KTRI - 1)
      UHFA(KTRI + 1) = URHF
      GO TO 177
  176 UHFA(KTRI+ 1) = 0.0
  177 KTRI = KTRI - 1
      IF(KTRI .GE. 1) GO TO 174
  178 IF(URHF .LE. 0.0) GO TO 179
      NRTRI = NCTRI
      IF(CONOPT(13) .EQ. 1) NRTRI = MXTRI
  179 NRTRI = NRTRI - 1
      UHFA(1) = 0.0
      IF(CONOPT(13) .NE. 1) GO TO 180
      NNSTRI = NCSTRI + 1
      UHFA(NNSTRI) = 0.0
  180 URHF = 0.0
  181 IF(SRX .LE. CSRX) SRX = CSRX
      RHF1 = TRHF - SRX*(TRHF - RHFO)
      RHFO = RHF1
      IF(RHFO .LT. RHFMC) RHFO = 0.0
      TFCFS = (4.0*RHF1 + CBF - HSE)*WCFS
      IF(CONOPT(13) .NE. 1) GO TO 182
      IF(CONOPT(12) .EQ. 1 .AND. PRD .NE. 4) GO TO 182
      CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXOPV,LSHFT,
     1  TFCFS)
      DATE = MOD(DAY,MDAY)
      IF(LSHFT) WRITE(6,6) DATE,HOUR,PRD,NCSTRI
    6 FORMAT(2X,I2,2X,I2,2X,I2,2X,20HHISTOGRAM CHANGES TO,1X,I2,1X,
     1  8HELEMENTS)
  182 CONTINUE
      IF(TFCFS .LE. 0.5*CHCAP) SRX = CSRX
      IF((TFCFS .GT. 0.5*CHCAP) .AND. (TFCFS .LT. 2.0*CHCAP)) SRX = CSRX
```

```
1   +(FSRX - CSRX)*((TFCFS - 0.5*CHCAP)/(1.5*CHCAP))**3
     IF(TFCFS .GT. 2.0*CHCAP) SRX = FSRX
     IF(TFCFS .LE. TFMAX) GO TO 183
     PRDF = PRD
     TDFP24 = HOUR
     IF(PRD .LE. 3) TDFP24 = (TDFP24 - 1.0) + 0.15*PRDF
     TFMAX = TFCFS
 183 ARHF = ARHF + RHF1
C   STORM OUTPUT REQUESTED BY CONOPT(1)
 184 IF(CONOPT(1) .NE. 1) GO TO 186
     IF(DAY .NE. CDSDR) GO TO 186
     IF(HOUR .EQ. 1 .AND. PRD .EQ. 1) WRITE(6,7)
   7 FORMAT(1H//,21X,19HRAINFALL DEPOSITION,12X,16HMOISTURE STORAGE,
    1  14X,17HSTREAMFLOW ORIGIN,6X,14HSTREAM OUTFLOW/2X,116HDY HR PD RA
    2IN   EUZS    ELZS    EIFS    EOFS         UZS      LZS      IFS       OFS        S
    3POF    SPIF    SPBF    SPTF   INCHES     CFS)
     DATE = MOD(DAY,MDAY)
     OFS = OFUS*FPER + OFUSIS*FIMP
     SPOF = OFR*FPER + OFRIS*FIMP + PPI*FWTR
     SPBF = 0.25*(CBF-HSE)
     SPTF = SPOR + SPBF
     SPOR = 0.0
     IF(RHFO .LE. 0.0) TFCFS = (CBF - HSE)*WCFS
     RSPTF = 0.25*TFCFS/WCFS
     WRITE(6,8) DATE,HOUR,PRD,PEP,PMEUZS,PMELZS,PMEIFS,PMEOFS,UZS,LZS
    1,IFS,OFS,SPOF,SPIF,SPBF,SPTF,RSPTF,TFCFS
   8 FORMAT(2X,I2,1X,I2,1X,I1,5(1X,F6.4),2X,4(F7.4),2X,5(1X,F6.4),1X,
    1  F7.1)
     IF(HOUR .EQ. 24 .AND. PRD .EQ. 4) GO TO 185
     GO TO 186
 185 NDSDP = NDSDP + 1
     IF(NDSDR .EQ. NDSDP) GO TO 186
     CALL DAYNXT(CDSDR,DPY)
 186 CONTINUE
     IF(VINTCR .LT. 0.25*VINTMR) VINTCR = VINTCR + DPET(DAY)/96.0
 187 CONTINUE
C   END OF 15 MINUTE LOOP
     IF(CONOPT(5) .NE. 1) GO TO 197
C   HOURLY OVERLAND FLOW AND RAINFALL SORTING
     IF(TOFR .LE. 0.0) GO TO 193
     KT20 = 20
 188 IF(KT20 .LT. 1) GO TO 192
     IF(TOFR .GT. T20OFH(KT20)) GO TO 189
     GO TO 190
 189 T20OFH(KT20+1) = T20OFH(KT20)
     GO TO 191
 190 T20OFH(KT20+1) = TOFR
     GO TO 193
 191 KT20 = KT20 - 1
     GO TO 188
 192 T20OFH(1) = TOFR
 193 IF(PRH .LE. 0.0) GO TO 197
     KT20 = 20
 194 IF(KT20 .LT. 1) GO TO 196
     IF(PRH .GT. T20PRH(KT20)) GO TO 195
     T20PRH(KT20 + 1) = PRH
     GO TO 197
 195 T20PRH(KT20+1) = T20PRH(KT20)
     KT20 = KT20 - 1
     GO TO 194
 196 T20PRH(1) = PRH
C   ADDING GROUNDWATER FLOW
 197 CBF = GWS*BFRL*(1.0 + BFNRL*BFNX)
     GWS = GWS - CBF
     AMBF = AMBF + CBF
     THGR = ARHF + CBF
     IF(HSE .GT. THGR) HSE = THGR
     AMSE = AMSE + HSE
     THSF(HOUR) = (THGR - HSE)*WCFS
     TDSF = TDSF + THSF(HOUR)
C
C   STORE SIMULATED HOURLY STREAM FLOWS
C   DRAINING OF UPPER ZONE STORAGE
     SINDEX=SINDEX+1
     THSFD(SINDEX,1)=THSF(HOUR)
306  UZINFX=(UZS/UZC) - (LZS/LZC)
     IF(UZINFX .LE. 0.0) GO TO 198
     LZSR = LZS/LZC
     UZINLZ = 0.003*BMIR*UZC*UZINFX**3.0
     IF(UZINLZ .GT. UZS) UZINLZ = UZS
     UZS = UZS - UZINLZ
     LZRX = 1.5*ABS(LZSR - 1.0) + 1.0
     FMR = (1.0/(1.0 + LZRX))**LZRX
     IF(LZS .LT. LZC) FMR = 1.0 - FMR*LZSR
     PGW = (1.0-FMR)*UZINLZ*(1.0 - SUBWF)*FPER
     PLZS = FMR*UZINLZ
     LZS = LZS + PLZS
     GWS = GWS + PGW
     BFNX = BFNX + PGW
```

```
C   4 PM ADJUSTMENTS OF VARIOUS VALUES                                  WCAS0354
  198 IF(HOUR .NE. 16) GO TO 202                                        WCAS0355
      AEX90 = 0.9*(AEX90 + PET)                                         WCAS0356
      AEX96 = 0.96*(AEX96 + PET)                                        WCAS0357
C   INFILTRATION CORRECTION                                             WCAS0358
      SIAM = (AEX96/AETX)**SIAC                                         WCAS0359
      IF(SIAM .LT. 0.33) SIAM = 0.33                                    WCAS0360
      BFNX = 0.97*BFNX                                                  WCAS0361
      IF(PET .EQ. 0.0) GO TO 202                                        WCAS0362
C   EVAP-TRANS LOSS FROM GROUNDWATER                                    WCAS0363
      GWET = GWS*GWETF*PET*FPER                                         WCAS0364
      GWS = GWS - GWET                                                  WCAS0365
      BFNX = BFNX - GWET                                                WCAS0366
      IF(BFNX .LT. 0.0) BFNX = 0.0                                      WCAS0367
      AMPET = AMPET + PET                                               WCAS0368
      IF(PET .GE. UZS) GO TO 199                                        WCAS0369
      UZS = UZS - PET                                                   WCAS0370
      AMNET = AMNET + PET                                               WCAS0371
      GO TO 202                                                         WCAS0372
  199 PET = PET - UZS                                                   WCAS0373
      AMNET = AMNET + UZS                                               WCAS0374
      UZS = 0.0                                                         WCAS0375
      LZSR = LZS/LZC                                                    WCAS0376
      IF(PET .GE. ETLF*LZSR) GO TO 200                                  WCAS0377
      SET = PET*(1.0 - PET/(2.0*ETLF*LZSR))                             WCAS0378
      GO TO 201                                                         WCAS0379
  200 SET = 0.5*ETLF*LZSR                                               WCAS0380
  201 LZS = LZS - SET                                                   WCAS0381
      AMNET = AMNET + SET                                               WCAS0382
  202 CONTINUE                                                          WCAS0383
C   END OF HOUR LOOP                                                    WCAS0384
      DSSF(DAY) = TDSF/24.0                                             WCAS0385
      IF(CONOPT(11) .EQ. 1) DSSF(DAY) = DSSF(DAY) + DDIW(DAY)           WCAS0386
  203 AMRTF = AMRTF + DRSF(DAY)                                         WCAS0387
      AMSTF = AMSTF + DSSF(DAY)                                         WCAS0388
      IF(CONOPT(6) .EQ. 1) EDLZS(DAY) = LZS                             WCAS0389
C   STORE ERRORS AND FLOW DURATION                                      WCAS0390
      IF(CONOPT(4) .NE. 1) GO TO 204                                    WCAS0391
      ERR = DSSF(DAY) - DRSF(DAY)                                       WCAS0392
      IF(DRSF(DAY) .LT. 1.0) KRFMI = 1.0                                WCAS0393
      IF(DRSF(DAY) .GT. 1.0) KRFMI= 2.0*ALOG(DRSF(DAY)) + 2.0           WCAS0394
      CRFMI(KRFMI) = CRFMI(KRFMI) + 1.0                                 WCAS0395
      SERR(KRFMI) = SERR(KRFMI) + ERR                                   WCAS0396
      SERA(KRFMI) = SERA(KRFMI) + ABS(ERR)                              WCAS0397
      SQER(KRFMI) = SQER(KRFMI) + ERR*ERR                               WCAS0398
      SESF(KRFMI) = 0.0                                                 WCAS0399
      IF(CRFMI(KRFMI) .GT. 1.0) SESF(KRFMI) = SQRT(ABS((SQER(KRFMI) -   WCAS0400
     1  SERR(KRFMI)**2/CRFMI(KRFMI))/(CRFMI(KRFMI) - 1.0)))             WCAS0401
  204 CONTINUE                                                          WCAS0402
      DATE=DAY                                                          WCAS0403
      IF ((MONTH.EQ.4).AND.(MDAY.EQ.31)) DATE=MOD(DAY,MDAY)             WCAS0404
      IF (MONTH.NE.4) DATE=MOD(DAY,MDAY)                                WCAS0405
      WRITE(6,9) DATE, (THSF(HOUR),HOUR=1,12)                           WCAS0406
    9 FORMAT(1H/,1X/,1X,I4,2X,2HAM,1X,6F8.1,3X,6F8.1)                   WCAS0407
      WRITE(6,10) (THSF(HOUR),HOUR=13,24), DSSF(DAY)                    WCAS0408
   10 FORMAT(1HJ,6X,2HPM,1X,6F8.1,3X,7F8.1)                             WCAS0409
      IF(TDFP24 .LT. 12.0) GO TO 205                                    WCAS0410
      TDFP12 = TDFP24 - 12.0                                            WCAS0411
   11 FORMAT(1H/,10X,8HMAXIMUM=,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,  WCAS0413
     1  4HP.M.)                                                         WCAS0414
      GO TO 206                                                         WCAS0415
  205 CONTINUE                                                          WCAS0000
   12 FORMAT(1H/,10X,8HMAXIMUM=,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,  WCAS0417
     1  4HA.M.)                                                         WCAS0418
  206 IF(CONOPT(7) .EQ. 1 .AND. SDEPTH .GT. 0.0) WRITE(6,13)DATE,       WCAS0419
     1SDEPTH,STMD,SAX,TANSM,SPLW                                        WCAS0420
   13 FORMAT(3X,I4,2X,7HSDEPTH=,F8.2,2X,5HSTMD=,F6.2,2X,4HSAX=,F6.2,    WCAS0421
     1  2X,6HTANSM=,F6.2,2X,5HSPLW=,F6.2)                               WCAS0422
      MAXI=MAXI+1                                                       WCAS0423
C   MONTHLY SUMMARY STORAGE                                             WCAS0424
      STMROS(MAXI,1)=DSSF(DAY)                                          WCAS0425
      TMSTF(MAXI,1)=AMSTF                                               WCAS0426
      AMSTF = 0.0                                                       WCAS0427
      TMRTF(MONTH) = AMRTF                                              WCAS0428
      AMRTF = 0.0                                                       WCAS0429
      EMBFNX(MAXI,1)=BFNX                                               WCAS0430
      TMPREC(MAXI,1)=AMPREC                                             WCAS0431
      AMPREC = 0.0                                                      WCAS0432
      TMRPM(MONTH) = AMRPM                                              WCAS0433
      AMRPM = 0.0                                                       WCAS0434
      TMBF(MAXI,1)=AMBF                                                 WCAS0435
      AMBF = 0.0                                                        WCAS0436
      TMIF(MAXI,1)=AMIF                                                 WCAS0437
      AMIF = 0.0                                                        WCAS0438
      TMSE(MAXI,1)=AMSE                                                 WCAS0439
      AMSE = 0.0                                                        WCAS0440
      TMPET(MONTH) = AMPET                                              WCAS0441
      AMPET = 0.0                                                       WCAS0442
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      TMNET(MONTH) = AMNET                                            WCAS0443
      AMNET = 0.0                                                     WCAS0444
      TMSNE(MONTH) = AMSNE                                            WCAS0445
      AMSNE = 0.0                                                     WCAS0446
      TMFSIL(MONTH) = AMFSIL                                          WCAS0447
      AMFSIL = 0.0                                                    WCAS0448
      EMGWS(MAXI,1)=GWS                                               WCAS0449
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZS/LZC)                       WCAS0450
      IF(UZC .LT. 0.25) UZC = 0.25                                    WCAS0451
      EMUZC(MAXI,1)=UZC                                               WCAS0452
      EMUZS(MAXI,1)=UZS                                               WCAS0453
      EMSIAM(MAXI,1)=SIAM                                             WCAS0454
      EMLZS(MAXI,1)=LZS                                               WCAS0455
      EMIFS(MAXI,1)=IFS                                               WCAS0456
220   CONTINUE                                                        WCAS0457
      IF  (MDAY.EQ.337) MDAY=59                                       WCAS0458
C                                                                     WCAS0459
C STORE MAXIMUM DAILY STREAM FLOW FOR YEAR                            WCAS0460
      TFMAXY(MAXI)=TFMAX                                              WCAS0461
3001  CONTINUE                                                        WCAS0462
      IF (DAY.EQ.EDATE) GO TO 221                                     WCAS0463
      CALL DAYNXT(DAY,DPY)                                            WCAS0464
      IF (DAY.GT.LDAY) MDAY=LDAY                                      WCAS0465
      IF (DAY.EQ.366)  MDAY=337                                       WCAS0466
      GO TO 152                                                       WCAS0467
C   END OF DAY LOOP                                                   WCAS0468
  221 CONTINUE                                                        WCAS0469
      RETURN                                                          WCAS0470
      END                                                             WCAS0471
```

ORIGINAL PAGE IS
OF POOR QUALITY

L-2

```
      SUBROUTINE   ZEROP(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
     2KFLAG,IDFLAG,
     1IENDEG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
     2  NCTRI,CTRI,FIRR,RICY,DPSE,BDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,
     3  XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
     A SATRI,UHFA,
     BMNRD,
     4  FWTR,VINTMR,BUZC,SUZC,LZC,FTLF,SUBWF,GWETF,SIAC,BMIR,
     5  BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,
     6  BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNPL,BFNHR,IFPRC,
     7  IFRL,LSHFT,NBTRI,ENTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAFT,EPER,
     8  TPLR,VINTCR,HSE,NRTRI,SPIF,CPF,SPOR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
     9  RHFO,LRHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SAPAX,SRX,VWIN,
     A  WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
     B  SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
     XDEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
     YTOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
     ZTSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
     WTORARY(1,6,1),
     CDRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)
      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
     1TMIF,TMPREC,TMSF,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
     2EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
     3TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,
     4TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
     5TRHF,
     5SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
C  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)
C     BASED ON STANFORD WATERSHED MODELS III & IV
      DIMENSION BTRI(99),CONOPT(16),CRFMI(22),CTRI(99),DDIW(366),
     1  DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
     2  DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
     3 EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
     4 EMUZC(15,3),EMUZS(15,3),EPCM(12),FTRP(15),MEDCY(12),MEDWY(12),
     5  RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),
     6 THSF(24),TITLE(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
     7 TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
     8 TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T200FH(21),T20PRH(21),
     . UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
     A PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)
      LOGICAL  LSHFT
      INTEGER   CDSDR,CN,CONOPT,DATE,DAY,DPY,EHSGD,HOUR,HRF,HRL,PDAY,
     1  PRD,RHPD,RHPH,RSBD,SGMD,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX
      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY
      INTEGER   DATES,EDATE
      DIMENSION RPLOTC(1832),RCOMMA(12087)
      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))
      REAL   IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT
      DATA   MEDCY/  0,           31,59,90,120,151,181,212,243,273,304,334/
      DATA   MEDWY/304,334,365,31,59,90,120,151,181,212,243,273           /
      REAL          MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD
      REAL   SSQM,SSQDI,SSQMI,VDRSF,VDSSF
      REAL   VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
     *              SMSQM
      REAL  MDRSF,MDSSF,MMRSF,MMSSF
      BACKSPACE 11
      READ  (11) (RPLOTC(I),I=1,1832)
      BACKSPACE 18
      READ(18) (RCOMMA(I),I=1,12087)
 150  CONTINUE
      DATE=PDATE
      DAY=DATES
      MDAY=PDAY
      SINDEX=0
      MAXI=0
      AMRPM = 0.0
      AMPREC = 0.0
      AMBF = 0.0
      AMSE = 0.0
      AMSTF = 0.0
      AMRTF = 0.0
      IF(DPY .EQ. 366) MEDWY(5)=366
 3    FORMAT(1H1,1X,'NO PRECIP HOURLY CFS VALUES',//,1X,4A4)
      WRITE(6,3)   MSBDIC
C  BEGIN DAY LOOP
      IMONTH=MONTH
  152 TDSF = 0.0
      IF (MONTH.NE.4) GO TO 148
      IF (MDAY.NE. 31) GO TO 149
 148  IF (DATE.GT.(MOD(DAY,MDAY))) IMONTH=MONTH+1
 149  CONTINUE
      PET=EPCM(IMONTH)*DPET(DAY)
      PETU = PET
      TFMAX = 0.0
C  EVAPOTRANSPIRATION ADJUSTMENTS
      IF(CONOPT(7) .NE. 1) GO TO 153
      IF(DMXT(DAY) - 4.0*ELDIF .LT. 40.0) PET = 0.0
      IF(SPTW .GT. SPTWCC) PET = FFOR*PET
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C  CALCULATICN OF SNOW EVAPORATION
      IF(DMNT(DAY) .GT. 32.0 .OR. SPTW .LE. DPSE(DAY)) GO TO 153
      SE = DPSE(DAY)
      AMSNE = AMSNE + SE
      SPTW = SPTW - SE
      IF(SFMD .GT. 0.0) SDEPTH = SDEPTH - SE/SFMD
  153 DO 202 HOUR = 1,24
  154 IF(HOUR .EQ. SGRT + 1) RGPM = DRGPM(DAY)
      IF(HOUR .EQ. 9) HSE = (FWTR*PET)/12.0
      IF(HOUR .EQ.21) HSE = 0.0
      PRH=0
      AMPREC = AMPREC  + PRH
C  ENTER SNOWMELT SUBROUTINE
      IF(CONOPT(7) .EQ. 1) CALL SNOMEL(BDDFSM,SPTWCC,SPM,ELDIF,DAY,
     1  SPBFLW, XDNFS,FFOR,FFSI,MRNSM,DSMGH,SDEPTH,STMD, PXCSA,HOUR,
     2  SAX,SOFRF,OFRFIS,SOFRFI,AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,
     3  WT4AM,WT4PM,ASM,ASMRG,SASFX,SARAX,DMXT,DMNT,RICY,FIRR)
  155 AMRPM = AMRPM + PRH
  156 TOFR = 0.0
      ARHF = 0.0
C  15 MINUTE ACCOUNTING AND ROUTING LOOP
      DO 187 PRD = 1,4
      PEBI = 0.0
      PPI = 0.0
      OFR = 0.0
      OFRIS = 0.0
      WI = 0.0
      WEIFS = 0.0
      PMEUZS = 0.0
      PMELZS = 0.0
      PMEIFS = 0.0
      PMEOFS = 0.0
      PEP = 0.25*PRH
      IF(CONOPT(2) .EQ. 1) CALL PREPRD(RGPM,DRHP,DAY,HOUR,DPY,PRD,PEP,
     1  PRH)
      IF(PEP .GT. 0.0) GO TO 157
      IF(OFUS .GT. 0.0) GO TO 159
      IF(IFS .GT. 0.0) GO TO 170
      IF(NRTRI .GT. 0) GO TO 172
      TRHF = 0.0
      IF(RHFO .GT. 0.0) GO TO 181
      GO TO 184
C  RAINFALL UPPER ZONE INTERACTION
  157 IF(PEP .GE. VINTCR) GO TO 158
      UZS = UZS + PEP*TPLR
      VINTCR = VINTCR - PEP
      PPI = 0.0
      PEBI = 0.0
      PMEUZS = PEP
      IF(OFUS .GT. 0.0) GO TO 159
      GO TO 170
  158 PPI = PEP - VINTCR
      UZS = UZS + VINTCR*TPLR
      VINTCR = 0.0
      LZSR = LZS/LZC
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZSR)
      IF(UZC .LT. 0.25) UZC = 0.25
      UZRX = 2.0*ABS(UZS/UZC - 1.0) + 1.0
      FMR = (1.0/(1.0 + UZRX))**UZRX
      IF(UZS .GT. UZC) FMR = 1.0 - FMR
      PEBI = PPI*FMR
      PMEUZS = PEP - PEBI
      UZS = UZS + PPI - PEBI
C  LOWER ZONE AND GROUNDWATER INFILTRATION
  159 LZSR = LZS/LZC
      EID = 4.0*LZSR
      IF(LZSR .LE. 1.0) GO TO 160
      EID = 4.0 + 2.0*(LZSR - 1.0)
      IF(LZSR .LE. 2.0) GO TO 160
      EID = 6.0
  160 PEBI = PEBI + OFUS
      CMIR = 0.25*SIAM*BMIR/(2.0**EID)
      CIVM = BIVF*2.0**LZSR
      IF(CIVM .LT. 1.0) CIVM = 1.0
      PEAI = PEBI*PEBI/(2.0*CMIR*CIVM)
      WI = PEBI*PEBI/(2.0*CMIR)
      IF(PEBI .GE. CMIR) WI = PEBI - 0.5*CMIR
      IF(PEBI .GE. CMIR*CIVM) PEAI = PEBI - 0.5*CMIR*CIVM
      WEIFS = WI - PEAI
      IF(PEBI .LE. OFUS) GO TO 161
      PMELZS = (PEBI - WI)*((PEBI - OFUS)/PEBI)
      PMEIFS = WEIFS*((PEBI - OFUS)/PEBI)
      PMEOFS = PEAI*((PEBI - OFUS)/PEBI)
  161 CONTINUE
      IF((PEAI - OFUS) .GT. 0.0) GO TO 162
      EQD = (OFUS + PEAI)/2.0
      GO TO 163
  162 EQD = EQDF*((PEAI - OFUS)**0.6)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
  163 IF((OFUS + PEAI) .GT. (2.0*EQD)) EQD = 0.5*(OFUS + PEAI)
      IF((OFUS + PEAI) .LE. 0.001) GO TO 164
      OFR = 0.25*OFRF*(((OFUS + PEAI)*0.5)**1.67)*((1.0 + 0.6*((OFUS +
     1  PEAI)/(2.0*EQD))**3.0)**1.67)
      IF(OFR .GT. (0.75*PEAI)) OFR = 0.75*PEAI
  164 IF(FIMP .EQ. 0.0) GO TO 168
  165 PEIS = PPI + OFUSIS
      IF((PEIS - OFUSIS) .GT. 0.0) GO TO 166
      EQDIS = (OFUSIS + PEIS)/2.0
      GO TO 167
  166 EQDIS = EQDFIS*((PEIS - OFUSIS)**0.6)
  167 IF((OFUSIS + PEIS) .GT. (2.0*EQDIS)) EQDIS = 0.5*(OFUSIS + PEIS)
      IF((OFUSIS + PEIS) .LE. 0.01) GO TO 168
      OFRIS = 0.25*OFRFIS*(((OFUSIS + PEIS)*0.5)**1.67)*((1.0 + 0.6*((
     1  OFUSIS + PEIS)/(2.0*EQDFIS))**3.0)**1.67)
      IF(OFRIS .GT. PEIS) OFRIS = PEIS
  168 TOFR = TOFR + FPER*OFR + FIMP*OFRIS + PPI*FWTR
      OFUSIS = PEIS - OFRIS
      OFUS = PEAI - OFR
      IF(OFUS .GE. 0.001) GO TO 169
      LZS = LZS + OFUS
      OFUS = 0.0
      OFRIS = OFRIS + OFUSIS
      OFUSIS = 0.0
  169 LZRX = 1.5*ABS(LZS/LZC - 1.0) + 1.0
      FMR = (1.0/(1.0 + LZRX))**LZPX
      IF(LZS .LT. LZC) FMR = 1.0 - FMR*(LZS/LZC)
      PLZS = FMR*(PEBI - WI)
      PGW = (1.0 -FMR)*(PEBI - WI)*(1.0 - SUBWF)*FPER
      GWS = GWS + PGW
      BFNX = BFNX + PGW
      LZS = LZS + PLZS
      IFS = IFS + WEIFS*FPER
  170 SPIF = IFRL*IFS
      AMIF = AMIF + SPIF
      IFS = IFS - SPIF
      IF(IFS .GE. 0.0001) GO TO 171
      LZS = LZS + IFS
      IFS = 0.0
  171 UHFA(1) = FPER*OFR + PPI*FWTR + FIMP*OFRIS + SPIF
      SPOR = UHFA(1)
C   ROUTING
  172 IF(CONOPT(12) .NE. 1) GO TO 173
      URHF = URHF + 0.25*UHFA(1)
      IF(PRD .NE. 4) GO TO 181
      UHFA(1) = URHF
  173 TRHF = 0.0
      KTRI = NCTRI
      IF(CONOPT(13) .EQ. 1) KTRI = NCSTRI
  174 URHF = UHFA(KTRI)
      IF(URHF .LE. 0.0) GO TO 176
  175 TRHF = TRHF + URHF*CTRI(KTRI)
      IF(CONOPT(13) .EQ. 1 .AND. LSHFT .AND. KTRI .GE. 2) TRHF = TRHF +
     1  URHF*SATRI(KTRI - 1)
      UHFA(KTRI + 1) = URHF
      GO TO 177
  176 UHFA(KTRI+ 1) = 0.0
  177 KTRI = KTRI - 1
      IF(KTRI .GE. 1) GO TO 174
  178 IF(URHF .LE. 0.0) GO TO 179
      NRTRI = NCTRI
      IF(CONOPT(13) .EQ. 1) NRTRI = MXTRI
  179 NRTRI = NRTRI - 1
      UHFA(1) = 0.0
      IF(CONOPT(13) .NE. 1) GO TO 180
      NNSTRI = NCSTRI + 1
      UHFA(NNSTRI) = 0.0
  180 URHF = 0.0
  181 IF(SRX .LE. CSRX) SRX = CSRX
      RHF1 = TRHF - SRX*(TRHF - RHFO)
      RHFO = RHF1
      IF(RHFO .LT. RHFMC) RHFO = 0.0
      TFCFS = (4.0*RHF1 + CBF - HSE)*WCFS
      IF(CONOPT(13) .NE. 1) GO TO 182
      IF(CONOPT(12) .EQ. 1 .AND. PRD .NE. 4) GO TO 182
      CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
     1  TFCFS)
      DATE = MOD(DAY,MDAY)
      IF(LSHFT) WRITE(6,6) DATE,HOUR,PRD,NCSTRI
    6 FORMAT(2X,I2,2X,I2,2X,I2,2X,20HHISTOGRAM CHANGES TO,1X,I2,1X,
     1  8HELEMENTS)
  182 CONTINUE
      IF(TFCFS .LE. 0.5*CHCAP) SRX = CSRX
      IF((TFCFS .GT. 0.5*CHCAP) .AND. (TFCFS .LT. 2.0*CHCAP)) SRX = CSRX
     1  +(FSRX - CSRX)*((TFCFS - 0.5*CHCAP)/(1.5*CHCAP))**3
      IF(TFCFS .GT. 2.0*CHCAP) SRX = FSRX
      IF(TFCFS .LE. TFMAX) GO TO 183
      PRDF = PRD
      TDFP24 = HOUR
```

```
      IF(PRD .LE. 3) TDFP24 = (TDFP24 - 1.0) + 0.15*PRDF
      TFMAX = TFCFS
  183 ARHF = ARHF + RHF1
C   STURM OUTPUT REQUESTED BY CONOPT(1)
  184 IF(CONOPT(1) .NE. 1) GO TO 186
      IF(DAY .NE. CDSDR) GO TO 186
      IF(HOUR .EQ. 1 .AND. PRD .EQ. 1) WRITE(6,7)
    7 FORMAT(1H/,21X,19HRAINFALL DISPOSITION,12X,16HMOISTURE STORAGE,
     1  14X,17HSTREAMFLOW ORIGIN,6X,14HSTREAM OUTFLOW/2X,116HDY HR PD RA
     2IN    EUZS    ELZS     EIFS     EOFS        UZS     LZS     IFS     OFS    S
     3POF    SPIF    SPBF    SPTF   INCHES     CFS)
      DATE = MOD(DAY,MDAY)
      OFS = OFUS*FPER + OFUSIS*FIMP
      SPOF = OFR*FPER + OFRIS*FIMP + PPI*FWTR
      SPBF = 0.25*(CBF-HSE)
      SPTF = SPDR + SPBF
      SPDR = 0.0
      IF(RHFO .LE. 0.0) TFCFS = (CBF - HSE)*WCFS
      RSPTF = 0.25*TFCFS/WCFS
      WRITE(6,8) DATE,HOUR,PRD,PFP,PMEUZS,PMELZS,PMEIFS,PMEOFS,UZS,LZS
     1,IFS,OFS,SPOF,SPIF,SPBF,SPTF,RSPTF,TFCFS
    8 FORMAT(2X,I2,1X,I2,1X,I1,5(1X,F6.4),2X,4(F7.4),2X,5(1X,F6.4),1X,
     1  F7.1)
      IF(HOUR .EQ. 24 .AND. PRD .EQ. 4) GO TO 185
      GO TO 186
  185 NDSDP = NDSDP + 1
      IF(NDSDR .EQ. NDSDP) GO TO 186
      CALL DAYNXT(CDSDR,DPY)
  186 CONTINUE
      IF(VINTCR .LT. 0.25*VINTMR) VINTCR = VINTCR + DPET(DAY)/96.0
  187 CONTINUE
C   END OF 15 MINUTE LOOP
      IF(CONOPT(5) .NE. 1) GO TO 197
C   HOURLY OVERLAND FLOW AND RAINFALL SORTING
      IF(TOFR .LE. 0.0) GO TO 193
      KT20 = 20
  188 IF(KT20 .LT. 1) GO TO 192
      IF(TOFR .GT. T20OFH(KT20)) GO TO 189
      GO TO 190
  189 T20OFH(KT20+1) = T20OFH(KT20)
      GO TO 191
  190 T20OFH(KT20+1) = TOFR
      GO TO 193
  191 KT20 = KT20 - 1
      GO TO 188
  192 T20OFH(1) = TOFR
  193 IF(PRH .LE. 0.0) GO TO 197
      KT20 = 20
  194 IF(KT20 .LT. 1) GO TO 196
      IF(PRH .GT. T20PRH(KT20)) GO TO 195
      T20PRH(KT20 + 1) = PRH
      GO TO 197
  195 T20PRH(KT20+1) = T20PRH(KT20)
      KT20 = KT20 - 1
      GO TO 194
  196 T20PRH(1) = PRH
C   ADDING GROUNDWATER FLOW
  197 CBF = GWS*BFRL*(1.0 + BFNRL*BFNX)
      GWS = GWS - CBF
      AMBF = AMBF + CBF
      THGR = ARHF + CBF
      IF(HSE .GT. THGR) HSE = THGR
      AMSE = AMSE + HSE
      THSF(HOUR) = (THGR - HSE)*WCFS
      TDSF = TDSF + THSF(HOUR)
C
C STORE SIMULATED HOURLY STREAM FLOWS
C   DRAINING OF UPPER ZONE STORAGE
      SINDEX=SINDEX+1
      THSFD(SINDEX,2)=THSF(HOUR)
306   UZINFX=(UZS/UZC) - (LZS/LZC)
      IF(UZINFX .LE. 0.0) GO TO 198
      LZSR = LZS/LZC
      UZINLZ = 0.003*BMIR*UZC*UZINFX**3.0
      IF(UZINLZ .GT. UZS) UZINLZ = UZS
      UZS = UZS - UZINLZ
      LZRX = 1.5*ABS(LZSR - 1.0) + 1.0
      FMR = (1.0/(1.0 + LZRX))**LZRX
      IF(LZS .LT. LZC) FMR = 1.0 - FMR*LZSR
      PGW = (1.0-FMR)*UZINLZ*(1.0 - SUBWF)*FPER
      PLZS = FMR*UZINLZ
      LZS = LZS + PLZS
      GWS = GWS + PGW
      BFNX = BFNX + PGW
C   4 PM ADJUSTMENTS OF VARIOUS VALUES
  198 IF(HOUR .NE. 16) GO TO 202
      AEX90 = 0.9*(AEX90 + PET)
      AEX96 = 0.96*(AEX96 + PET)
C   INFILTRATION CORRECTION
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SIAM = (AEX96/AETX)**SIAC
      IF(SIAM .LT. 0.33) SIAM = 0.33
      BFNX = 0.97*BFNX
      IF(PET .EQ. 0.0) GO TO 202
C  EVAP-TRANS LOSS FROM GROUNDWATER
      GWET = GWS*GWETF*PET*FPER
      GWS = GWS - GWET
      BFNX = BFNX - GWET
      IF(BFNX .LT. 0.0) BFNX = 0.0
      AMPET = AMPET + PET
      IF(PET .GE. UZS) GO TO 199
      UZS = UZS - PET
      AMNET = AMNET + PET
      GO TO 202
  199 PET = PET - UZS
      AMNET = AMNET + UZS
      UZS = 0.0
      LZSR = LZS/LZC
      IF(PET .GE. ETLF*LZSR) GO TO 200
      SET = PET*(1.0 - PET/(2.0*ETLF*LZSR))
      GO TO 201
  200 SET = 0.5*ETLF*LZSR
  201 LZS = LZS - SET
      AMNET = AMNET + SET
  202 CONTINUE
C  END OF HOUR LOOP
      DSSF(DAY) = TDSF/24.0
      IF(CONOPT(11) .EQ. 1) DSSF(DAY) = DSSF(DAY) + DDIW(DAY)
  203 AMRTF = AMRTF + DRSF(DAY)
      AMSTF = AMSTF + DSSF(DAY)
      IF(CONOPT(6) .EQ. 1) EDLZS(DAY) = LZS
C  STORE ERRORS AND FLOW DURATION
      IF(CONOPT(4) .NE. 1) GO TO 204
      ERR = DSSF(DAY) - DRSF(DAY)
      IF(DRSF(DAY) .LT. 1.0) KRFMI = 1.0
      IF(DRSF(DAY) .GT. 1.0) KRFMI= 2.0*ALOG(DRSF(DAY)) + 2.0
      CRFMI(KRFMI) = CRFMI(KRFMI) + 1.0
      SERR(KRFMI) = SERR(KRFMI) + ERR
      SERA(KRFMI) = SERA(KRFMI) + ABS(ERR)
      SQER(KRFMI) = SQER(KRFMI) + ERR*ERR
      SESF(KRFMI) = 0.0
      IF(CRFMI(KRFMI) .GT. 1.0) SESF(KRFMI) = SQRT(ABS((SQER(KRFMI) -
     1  SERR(KRFMI)**2/CRFMI(KRFMI))/(CRFMI(KRFMI) - 1.0)))
204   CONTINUE
      DATE=DAY
      IF ((MONTH.EQ.4).AND.(MDAY.EQ.31)) DATE=MOD(DAY,MDAY)
      IF (MONTH.NE.4) DATE=MOD(DAY,MDAY)
      WRITE(6,9) DATE, (THSF(HOUR),HOUR=1,12)
    9 FORMAT(1H/,1X/,1X,I4,2X,2HAM,1X,6F8.1,3X,6F8.1)
      WRITE(6,10) (THSF(HOUR),HOUR=13,24), DSSF(DAY)
   10 FORMAT(1HJ,6X,2HPM,1X,6F8.1,3X,7F8.1)
      IF(TDFP24 .LT. 12.0) GO TO 205
      TDFP12 = TDFP24 - 12.0
   11 FORMAT(1H/,10X,8HMAXIMUM=,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,
     1  4HP.M.)
      GO TO 206
205   CONTINUE
   12 FORMAT(1H/,10X,8HMAXIMUM=,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,
     1  4HA.M.)
  206 IF(CONOPT(7) .EQ. 1 .AND. SDEPTH .GT. 0.0) WRITE(6,13)DATE,
     1SDEPTH,STMD,SAX,TANSM,SPLW
   13 FORMAT(3X,I4,2X,7HSDEPTH=,F8.2,2X,5HSTMD=,F6.2,2X,4HSAX=,F6.2,
     1  2X,6HTANSM=,F6.2,2X,5HSPLW=,F6.2)
      MAXI=MAXI+1
C  MONTHLY SUMMARY STORAGE
      STMROS(MAXI,2)=DSSF(DAY)
      TMSTF(MAXI,2)=AMSTF
      AMSTF = 0.0
      TMRTF(MONTH) = AMRTF
      AMRTF = 0.0
      EMBFNX(MAXI,2)=BFNX
      TMPREC(MAXI,2)=AMPREC
      AMPREC = 0.0
      TMRPM(MONTH) = AMRPM
      AMRPM = 0.0
      TMBF(MAXI,2)=AMBF
      AMBF = 0.0
      TMIF(MAXI,2)=AMIF
      AMIF = 0.0
      TMSE(MAXI,2)=AMSE
      AMSE = 0.0
      TMPET(MONTH) = AMPET
      AMPET = 0.0
      TMNET(MONTH) = AMNET
      AMNET = 0.0
      TMSNE(MONTH) = AMSNE
      AMSNE = 0.0
      TMFSIL(MONTH) = AMFSIL
      AMFSIL = 0.0
```

ORIGINAL PAGE IS OF POOR QUALITY

```
      EMGWS(MAXI,2)=GWS
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZS/LZC)
      IF(UZC .LT. 0.25) UZC = 0.25
      EMUZC(MAXI,2)=UZC
      EMUZS(MAXI,2)=UZS
      EMSIAM(MAXI,2)=SIAM
      EMLZS(MAXI,2)=LZS
      EMIFS(MAXI,2)=IFS
220   CONTINUE
      IF  (MDAY.EQ.337) MDAY=59
C
C STORE MAXIMUM DAILY STREAM FLOW FOR YEAR
      TFMAXY(MAXI)=TFMAX
3001  CONTINUE
      IF (DAY.EQ.EDATE)  GO TO 221
      CALL DAYNXT(DAY,DPY)
      IF (DAY.GT.LDAY) MDAY=LDAY
      IF (DAY.EQ.366)  MDAY=337
      GO TO 152
C  END OF DAY LOOP
  221 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE FORCST(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY) FCST0001
       COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,FCST0002
     2KFLAG,IDFLAG,                                                      FCST0003
     1IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,        FCST0004
     2  NCTRI,CTRI,FIRR,RICY,DPSC,BDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,        FCST0005
     3  XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,KGPMB,AREA,FIMP,          FCST0006
     A SATRI,UHFA,                                                       FCST0007
     BMNRD,                                                              FCST0008
     4  FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,            FCST0009
     5  BIVF,CFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,           FCST0010
     6  BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHPC,BFRL,BFNRL,BFNHR,IFPRC,    FCST0011
     7  IFRL,LSHFT,NFTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAET,EPER,       FCST0012
     8  TPLR,VINTCR,HSE,NRTRI,SPIF,CRF,SPDR,UFUS,UFUSIS,OFR,OFRIS,PEIS,  FCST0013
     9  RHFO,URHF,AMIF,AMJET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,    FCST0014
     A  WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,            FCST0015
     B  SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,  FCST0016
     XDEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),                     FCST0017
     YTOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),                 FCST0018
     ZTSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),                   FCST0019
     WTORARY(1,6,1),                                                     FCST0020
     CDRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)             FCST0021
       COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,    FCST0022
     1TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,FCST0023
     2EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,FCST0024
     3TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,                                FCST0025
     4TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,         FCST0026
     5TRHF,                                                              FCST0027
     5SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS                      FCST0028
C   KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)                   FCST0029
C       BASED ON STANFORD WATERSHED MODELS III & IV                      FCST0030
       DIMENSION BTRI(99),CONOPT(16),CRFMI(22),CTRI(99),DDIW(366),       FCST0031
     1  DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),       FCST0032
     2  DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),         FCST0033
     3 EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),    FCST0034
     4 EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),    FCST0035
     5  RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),     FCST0036
     6 THSF(24),TITLE(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),    FCST0037
     7 TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),FCST0038
     8 TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T200FH(21),T20PRH(21),         FCST0039
     9 UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),           FCST0040
     A PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)                               FCST0041
       LOGICAL  LSHFT                                                    FCST0042
       INTEGER  CDSDR,CN,CONOPT,DATE,DAY,DPY,EHSGD,HOUR,HRF,HRL,PDAY,    FCST0043
     1  PRD,RHPD,RHPH,RSBD,SGMD,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX      FCST0044
       INTEGER TUMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY                 FCST0045
       INTEGER  DATES,FDATE,SINDET,MPDAY(15)                             FCST0046
       DIMENSION RPLOTC(1832),RCOMMA(12087)                              FCST0047
       EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))                     FCST0048
       REAL   IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNPD,MRNSM,NHPT FCST0049
       DATA   MEDCY/  0,        31,59,90,120,151,181,212,243,273,304,334/ FCST0050
       DATA   MEDWY/304,334,365,31,59,90,120,151,181,212,243,273       / FCST0051
       REAL       MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD                       FCST0052
       REAL   SSQM,SSQDI,SSQMI,VDRSF,VDSSF                               FCST0053
       REAL   VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,   FCST0054
     *            SMSQM                                                  FCST0055
       REAL   MDRSF,MDSSF,MMRSF,MMSSF                                    FCST0056
       BACKSPACE 11                                                      FCST0057
       READ  (11) (RPLOTC(I),I=1,1832)                                   FCST0058
       BACKSPACE 18                                                      FCST0059
       READ(18) (RCOMMA(I),I=1,12087)                                    FCST0060
150    CONTINUE                                                          FCST0061
       DATE=PDATE                                                        FCST0062
       DAY=DATES                                                         FCST0063
       MDAY=PDAY                                                         FCST0064
       SINDEX=0                                                          FCST0065
       MAXI=0                                                            FCST0066
       AMRPM = 0.0                                                       FCST0067
       AMPREC = 0.0                                                      FCST0068
       AMBF = 0.0                                                        FCST0069
       AMSE = 0.0                                                        FCST0070
       AMSTF = 0.0                                                       FCST0071
       AMRTF = 0.0                                                       FCST0072
       IF(DPY .EQ. 366) MEDWY(5)=366                                     FCST0073
3      FORMAT(1H1,1X,'FORECAST CASE HOURLY CFS VALUES',//,1X,4A4)        FCST0000
       WRITE(6,3)  MSBDIC                                                FCST0075
C   BEGIN DAY LOOP                                                       FCST0076
       IMONTH=MONTH                                                      FCST0077
   152 TDSF = 0.0                                                        FCST0078
       IF (MONTH.NE.4) GO TO 148                                         FCST0079
       IF (MDAY.NE. 31) GO TO 149                                        FCST0080
148    IF (DATE.GT.(MOD(DAY,MDAY))) IMONTH=MONTH+1                       FCST0081
149    CONTINUE                                                          FCST0082
       PET=EPCM(IMONTH)*DPET(DAY)                                        FCST0083
       PETU = PET                                                        FCST0084
       TFMAX = 0.0                                                       FCST0085
C   EVAPOTRANSPIRATION ADJUSTMENTS                                       FCST0086
       IF(CONOPT(7) .NE. 1) GO TO 153                                    FCST0087
       IF(DMXT(DAY) - 4.0*ELDIF .LT. 40.0) PET = 0.0                     FCST0088
       IF(SPTW .GT. SPTWCC) PET = FFOR*PET                               FCST0089
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C  CALCULATION OF SNOW EVAPORATION
      IF(DMNT(DAY) .GT. 32.0 .OR. SPTW .LE. DPSE(DAY)) GO TO 153
      SE = DPSE(DAY)
      AMSNE = AMSNE + SE
      SPTW = SPTW - SE
      IF(SFMD .GT. 0.0) SDEPTH = SDEPTH - SE/SFMD
  153 DO 202 HOUR = 1,24
      IF((NSGRD .EQ. 0) .AND. (DRHP(DAY,HOUR) .NE. 0.0)  .AND. (PET .EQ.
     1  PETU) .AND. (CONOPT(3) .EQ. 1)) PET = 0.5*PET
  154 IF(HOUR .EQ. SGRT + 1) RGPM = DRGPM(DAY)
      IF(HOUR .EQ. 9) HSE = (FWTR*PET)/12.0
      IF(HOUR .EQ.21) HSE = 0.0
      PRH = RGPM*DRHP(DAY,HOUR)
      AMPREC = AMPREC  + PRH
C  ENTER SNOWMELT SUBROUTINE
      IF(CONOPT(7) .EQ. 1) CALL SNOMEL(BDDFSM,SPTWCC,SPM,ELDIF,DAY,
     1  SPBFLW, XDNFS,FFOR,FFSI,MRNSM,DSMGH,SDEPTH,STMD, PXCSA,HOUR,
     2  SAX,SOFRF,CFRFIS,SOFRFI,AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,CFRF,
     3  WT4AM,WT4PM,ASM,ASMRG,SASFX,SARAX,DMXT,DMNT,RICY,FIRR)
  155 AMRPM = AMRPM + PRH
  156 TOFR = 0.0
      ARHF = 0.0
C  15 MINUTE ACCOUNTING AND ROUTING LOOP
      DO 187 PRD = 1,4
      PEBI = 0.0
      PPI = 0.0
      OFR = 0.0
      OFRIS = 0.0
      WI = 0.0
      WEIFS = 0.0
      PMEUZS = 0.0
      PMELZS = 0.0
      PMEIFS = 0.0
      PMEOFS = 0.0
      PEP = 0.25*PRH
      IF(CONOPT(2) .EQ. 1) CALL PREPRD(RGPM,DRHP,DAY,HOUR,DPY,PRD,PEP,
     1  PRH)
      IF(PEP .GT. 0.0) GO TO 157
      IF(OFUS .GT. 0.0) GO TO 159
      IF(IFS .GT. 0.0) GO TO 170
      IF(NRTRI .GT. 0) GO TO 172
      TRHF = 0.0
      IF(RHFO .GT. 0.0) GO TO 181
      GO TO 184
C  RAINFALL UPPER ZONE INTERACTION
  157 IF(PEP .GE. VINTCR) GO TO 158
      UZS = UZS + PEP*TPLR
      VINTCR = VINTCR - PEP
      PPI = 0.0
      PEBI = 0.0
      PMEUZS = PEP
      IF(OFUS .GT. 0.0) GO TO 159
      GO TO 170
  158 PPI = PEP - VINTCR
      UZS = UZS + VINTCR*TPLR
      VINTCR = 0.0
      LZSR = LZS/LZC
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZSR)
      IF(UZC .LT. 0.25) UZC = 0.25
      UZRX = 2.0*ABS(UZS/UZC - 1.0) + 1.0
      FMR = (1.0/(1.0 + UZRX))**UZPX
      IF(UZS .GT. UZC) FMR = 1.0 - FMR
      PEBI = PPI*FMR
      PMEUZS = PEP - PEBI
      UZS = UZS + PPI - PEBI
C  LOWER ZONE AND GROUNDWATER INFILTRATION
  159 LZSR = LZS/LZC
      EID = 4.0*LZSR
      IF(LZSR .LE. 1.0) GO TO 160
      EID = 4.0 + 2.0*(LZSR - 1.0)
      IF(LZSR .LE. 2.0) GO TO 160
      EID = 6.0
  160 PEBI = PEBI + OFUS
      CMIR = 0.25*SIAM*BMIR/(2.0**EID)
      CIVM = BIVF*2.0**LZSR
      IF(CIVM .LT. 1.0) CIVM = 1.0
      PEAI = PEBI*PEBI/(2.0*CMIR*CIVM)
      WI = PEBI*PEBI/(2.0*CMIR)
      IF(PEBI .GE. CMIR) WI = PEBI - 0.5*CMIR
      IF(PEBI .GE. CMIR*CIVM) PEAI = PEBI - 0.5*CMIR*CIVM
      WEIFS = WI - PEAI
      IF(PEBI .LE. OFUS) GO TO 161
      PMELZS = (PEBI - WI)*((PEBI - OFUS)/PEBI)
      PMEIFS = WEIFS*((PEBI - OFUS)/PEBI)
      PMEOFS = PEAI*((PEBI - OFUS)/PEBI)
  161 CONTINUE
      IF((PEAI - OFUS) .GT. 0.0) GO TO 162
      EQD = (OFUS + PEAI)/2.0
      GO TO 163
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
162 EQD = EQDF*((PEAI - OFUS)**0.6)
163 IF((OFUS + PEAI) .GT. (2.0*EQD)) EQD = 0.5*(OFUS + PEAI)
    IF((OFUS + PEAI) .LE. 0.001) GO TO 164
    OFR = 0.25*OFRF*(((OFUS + PEAI)*0.5)**1.67)*((1.0 + 0.6*((OFUS +
   1  PEAI)/(2.0*EQD))**3.0)**1.67)
    IF(OFR .GT. (0.75*PEAI)) OFR = 0.75*PEAI
164 IF(FIMP .EQ. 0.0) GO TO 168
165 PEIS = PPI + OFUSIS
    IF((PEIS - OFUSIS) .GT. 0.0) GO TO 166
    EQDIS = (OFUSIS + PEIS)/2.0
    GO TO 167
166 EQDIS = EQDFIS*((PEIS - OFUSIS)**0.6)
167 IF((OFUSIS + PEIS) .GT. (2.0*EQDIS)) EQDIS = 0.5*(OFUSIS + PEIS)
    IF((OFUSIS + PEIS) .LE. 0.01) GO TO 168
    OFRIS = 0.25*OFRFIS*(((OFUSIS + PEIS)*0.5)**1.67)*((1.0 + 0.6*((
   1  OFUSIS + PEIS)/(2.0*EQDFIS))**3.0)**1.67)
    IF(OFRIS .GT. PEIS) OFRIS = PEIS
168 TOFR = TOFR + FPER*OFR + FIMP*OFRIS + PPI*FWTR
    OFUSIS = PEIS - OFRIS
    OFUS = PEAI - OFR
    IF(OFUS .GE. 0.001) GO TO 169
    LZS = LZS + OFUS
    OFUS = 0.0
    OFRIS = OFRIS + OFUSIS
    OFUSIS = 0.0
169 LZRX = 1.5*ABS(LZS/LZC - 1.0) + 1.0
    FMR = (1.0/(1.0 + LZRX))**LZRX
    IF(LZS .LT. LZC) FMR = 1.0 - FMR*(LZS/LZC)
    PLZS = FMR*(PEBI - WI)
    PGW = (1.0 -FMR)*(PEBI - WI)*(1.0 - SUBWF)*FPER
    GWS = GWS + PGW
    BFNX = BFNX + PGW
    LZS = LZS + PLZS
    IFS = IFS + WEIFS*FPER
170 SPIF = IFRL*IFS
    AMIF = AMIF + SPIF
    IFS = IFS - SPIF
    IF(IFS .GE. 0.0001) GO TO 171
    LZS = LZS + IFS
    IFS = 0.0
171 UHFA(1) = FPER*OFR + PPI*FWTR + FIMP*OFRIS + SPIF
    SPDR = UHFA(1)
C  ROUTING
172 IF(CONOPT(12) .NE. 1) GO TO 173
    URHF = URHF + 0.25*UHFA(1)
    IF(PRD .NE. 4) GO TO 181
    UHFA(1) = URHF
173 TRHF = 0.0
    KTRI = NCTRI
    IF(CONOPT(13) .EQ. 1) KTRI = NCSTRI
174 URHF = UHFA(KTRI)
    IF(URHF .LE. 0.0) GO TO 176
175 TRHF = TRHF + URHF*CTRI(KTRI)
    IF(CONOPT(13) .EQ. 1 .AND. LSHFT .AND. KTRI .GE. 2) TRHF = TRHF +
   1  URHF*SATRI(KTRI - 1)
    UHFA(KTRI + 1) = URHF
    GO TO 177
176 UHFA(KTRI+ 1) = 0.0
177 KTRI = KTRI - 1
    IF(KTRI .GE. 1) GO TO 174
178 IF(URHF .LE. 0.0) GO TO 179
    NRTRI = NCTRI
    IF(CONOPT(13) .EQ. 1) NRTRI = MXTRI
179 NRTRI = NRTRI - 1
    UHFA(1) = 0.0
    IF(CONOPT(13) .NE. 1) GO TO 180
    NNSTRI = NCSTRI + 1
    UHFA(NNSTRI) = 0.0
180 URHF = 0.0
181 IF(SRX .LE. CSRX) SRX = CSRX
    RHF1 = TRHF - SRX*(TRHF - RHFO)
    RHFO = RHF1
    IF(RHFO .LT. RHFMC) RHFO = 0.0
    TFCFS = (4.0*RHF1 + CBF - HSE)*WCFS
    IF(CCNOPT(13) .NE. 1) GO TO 182
    IF(CCNOPT(12) .EQ. 1 .AND. PRD .NE. 4) GO TO 182
    CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
   1  TFCFS)
    DATE = MOD(DAY,MDAY)
    IF(LSHFT) WRITE(6,6) DATE,HOUR,PRD,NCSTRI
  6 FORMAT(2X,I2,2X,I2,2X,I2,2X,20HHISTOGRAM CHANGES TO,1X,I2,1X,
   1  8HELEMENTS)
182 CONTINUE
    IF(TFCFS .LE. 0.5*CHCAP) SRX = CSRX
    IF((TFCFS .GT. 0.5*CHCAP) .AND. (TFCFS .LT. 2.0*CHCAP)) SRX = CSRX
   1  +(FSRX - CSRX)*((TFCFS - 0.5*CHCAP)/(1.5*CHCAP))**3
    IF(TFCFS .GT. 2.0*CHCAP) SRX = FSRX
    IF(TFCFS .LE. TFMAX) GO TO 183
    PRDF = PRD
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
      TDFP24 = HOUR
      IF(PRD .LE. 3) TDFP24 = (TDFP24 - 1.0) + 0.15*PRDF
      TFMAX = TFCFS
  183 ARHF = ARHF + RHF1
C  STORM OUTPUT REQUESTED BY CONOPT(1)
  184 IF(CONOPT(1) .NE. 1) GO TO 186
      IF(DAY .NE. CDSDR) GO TO 186
      IF(HOUR .EQ. 1 .AND. PRD .EQ. 1) WRITE(6,7)
    7 FORMAT(1H//,21X,19HRAINFALL DEPOSITION,12X,16HMOISTURE STORAGE,
     1  14X,17HSTREAMFLOW ORIGIN,6X,14HSTREAM OUTFLOW/2X,116HDY HR PD RA
     2IN   EUZS   ELZS   EIFS   EOFS        UZS    LZS    IFS    OFS    S
     3POF   SPIF   SPBF   SPTF  INCHES    CFS)
      DATE = MOD(DAY,MDAY)
      OFS = OFUS*FPER + OFUSIS*FIMP
      SPOF = OFR*FPER + OFRIS*FIMP + PPI*FWTR
      SPBF = 0.25*(CBF-HSE)
      SPTF = SPDR + SPBF
      SPDR = 0.0
      IF(RHFO .LE. 0.0) TFCFS = (CBF - HSE)*WCFS
      RSPTF = 0.25*TFCFS/WCFS
      WRITE(6,8) DATE,HOUR,PRD,PEP,PMEUZS,PMELZS,PMEIFS,PMEOFS,UZS,LZS
     1,IFS,OFS,SPOF,SPIF,SPBF,SPTF,RSPTF,TFCFS
    8 FORMAT(2X,I2,1X,I2,1X,I1,5(1X,F6.4),2X,4(F7.4),2X,5(1X,F6.4),1X,
     1  F7.1)
      IF(HOUR .EQ. 24 .AND. PRD .EQ. 4) GO TO 185
      GO TO 186
  185 NDSDP = NDSDP + 1
      IF(NDSDR .EQ. NDSDP) GO TO 186
      CALL DAYNXT(CDSDR,DPY)
  186 CONTINUE
      IF(VINTCR .LT. 0.25*VINTMR) VINTCR = VINTCR + DPET(DAY)/96.0
  187 CONTINUE
C  END OF 15 MINUTE LOOP
      IF(CONOPT(5) .NE. 1) GO TO 197
C  HOURLY OVERLAND FLOW AND RAINFALL SORTING
      IF(TOFR .LE. 0.0) GO TO 193
      KT20 = 20
  188 IF(KT20 .LT. 1) GO TO 192
      IF(TOFR .GT. T20OFH(KT20)) GO TO 189
      GO TO 190
  189 T20OFH(KT20+1) = T20OFH(KT20)
      GO TO 191
  190 T20OFH(KT20+1) = TOFR
      GO TO 193
  191 KT20 = KT20 - 1
      GO TO 188
  192 T20OFH(1) = TOFR
  193 IF(PRH .LE. 0.0) GO TO 197
      KT20 = 20
  194 IF(KT20 .LT. 1) GO TO 196
      IF(PRH .GT. T20PRH(KT20)) GO TO 195
      T20PRH(KT20 + 1) = PRH
      GO TO 197
  195 T20PRH(KT20+1) = T20PRH(KT20)
      KT20 = KT20 - 1
      GO TO 194
  196 T20PRH(1) = PRH
C  ADDING GROUNDWATER FLOW
  197 CBF = GWS*BFRL*(1.0 + BFNRL*BFNX)
      GWS = GWS - CBF
      AMBF = AMBF + CBF
      THGR = ARHF + CBF
      IF(HSE .GT. THGR) HSE = THGR
      AMSE = AMSE + HSE
      THSF(HOUR) = (THGR - HSE)*WCFS
      TDSF = TDSF + THSF(HOUR)
C
C STORE SIMULATED HOURLY STREAM FLOWS
C  DRAINING OF UPPER ZONE STORAGE
      SINDEX=SINDEX+1
      THSFD(SINDEX,3)=THSF(HOUR)
306   UZINFX=(UZS/UZC) - (LZS/LZC)
      IF(UZINFX .LE. 0.0) GO TO 198
      LZSR = LZS/LZC
      UZINLZ = 0.003*BMIR*UZC*UZINFX**3.0
      IF(UZINLZ .GT. UZS) UZINLZ = UZS
      UZS = UZS - UZINLZ
      LZRX = 1.5*ABS(LZSR - 1.0) + 1.0
      FMR = (1.0/(1.0 + LZRX))**LZRX
      IF(LZS .LT. LZC) FMR = 1.0 - FMR*LZSR
      PGW = (1.0-FMR)*UZINLZ*(1.0 - SUBWF)*FPER
      PLZS = FMR*UZINLZ
      LZS = LZS + PLZS
      GWS = GWS + PGW
      BFNX = BFNX + PGW
C  4 PM ADJUSTMENTS OF VARIOUS VALUES
  198 IF(HOUR .NE. 16) GO TO 202
      AEX90 = 0.9*(AEX90 + PET)
      AEX96 = 0.96*(AEX96 + PET)
```

```
C   INFILTRATION CORRECTION                                            FCST0357
      SIAM = (AEX96/AETX)**SIAC                                         FCST0358
      IF(SIAM .LT. 0.33) SIAM = 0.33                                    FCST0359
      BFNX = 0.97*BFNX                                                  FCST0360
      IF(PET .EQ. 0.0) GO TO 202                                        FCST0361
C   EVAP-TRANS LOSS FROM GROUNDWATER                                   FCST0362
      GWET = GWS*GWETF*PET*FPER                                         FCST0363
      GWS = GWS - GWET                                                  FCST0364
      BFNX = BFNX - GWET                                                FCST0365
      IF(BFNX .LT. 0.0) BFNX = 0.0                                      FCST0366
      AMPET = AMPET + PET                                               FCST0367
      IF(PET .GE. UZS) GO TO 199                                        FCST0368
      UZS = UZS - PET                                                   FCST0369
      AMNET = AMNET + PET                                               FCST0370
      GO TO 202                                                         FCST0371
  199 PET = PET - UZS                                                   FCST0372
      AMNET = AMNET + UZS                                               FCST0373
      UZS = 0.0                                                         FCST0374
      LZSR = LZS/LZC                                                    FCST0375
      IF(PET .GE. ETLF*LZSR) GO TO 200                                  FCST0376
      SET = PET*(1.0 - PET/(2.0*ETLF*LZSR))                             FCST0377
      GO TO 201                                                         FCST0378
  200 SET = 0.5*ETLF*LZSR                                               FCST0379
  201 LZS = LZS - SET                                                   FCST0380
      AMNET = AMNET + SET                                               FCST0381
  202 CONTINUE                                                          FCST0382
C   END OF HOUR LOOP                                                   FCST0383
      DSSF(DAY) = TDSF/24.0                                             FCST0384
      IF(CONOPT(11) .EQ. 1) DSSF(DAY) = DSSF(DAY) + DDIW(DAY)           FCST0385
  203 AMRTF = AMRTF + DRSF(DAY)                                         FCST0386
      AMSTF = AMSTF + DSSF(DAY)                                         FCST0387
      IF(CONOPT(6) .EQ. 1) EDLZS(DAY) = LZS                             FCST0388
C   STORE ERRORS AND FLOW DURATION                                     FCST0389
      IF(CONOPT(4) .NE. 1) GO TO 204                                    FCST0390
      ERR = DSSF(DAY) - DRSF(DAY)                                       FCST0391
      IF(DRSF(DAY) .LT. 1.0) KREMI = 1.0                                FCST0392
      IF(DRSF(DAY) .GT. 1.0) KREMI= 2.0*ALOG(DRSF(DAY)) + 2.0           FCST0393
      CREMI(KREMI) = CREMI(KREMI) + 1.0                                 FCST0394
      SERR(KREMI) = SERR(KREMI) + ERR                                   FCST0395
      SERA(KREMI) = SERA(KREMI) + ABS(ERR)                              FCST0396
      SQER(KREMI) = SQER(KREMI) + ERR*ERR                               FCST0397
      SESF(KREMI) = 0.0                                                 FCST0398
      IF(CREMI(KREMI) .GT. 1.0) SESF(KREMI) = SQRT(ABS((SQER(KREMI) -   FCST0399
     1  SERR(KREMI)**2/CREMI(KREMI))/(CREMI(KREMI) - 1.0)))             FCST0400
 204  CONTINUE                                                          FCST0401
      DATE=DAY                                                          FCST0402
      IF ((MONTH.EQ.4).AND.(MDAY.EQ.31)) DATE=MOD(DAY,MDAY)             FCST0403
      IF (MONTH.NE.4) DATE=MOD(DAY,MDAY)                                FCST0404
      WRITE(6,9) DATE, (THSF(HOUR),HOUR=1,12)                           FCST0405
    9 FORMAT(1H/,1X/,1X,I4,2X,2HAM,1X,6F8.1,3X,6F8.1)                   FCST0406
      WRITE(6,10) (THSF(HOUR),HOUR=13,24), DSSF(DAY)                    FCST0407
   10 FORMAT(1HJ,6X,2HPM,1X,6F8.1,3X,7F8.1)                             FCST0408
      IF(TDFP24 .LT. 12.0) GO TO 205                                    FCST0409
      TDFP12 = TDFP24 - 12.0                                            FCST0410
   11 FORMAT(1H/,10X,8HMAXIMUM=,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,  FCST0412
     1  4HP.M.)                                                         FCST0413
      GO TO 206                                                         FCST0414
 205  CONTINUE                                                          FCST0000
   12 FORMAT(1H/,10X,8HMAXIMUM=,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,  FCST0416
     1  4HA.M.)                                                         FCST0417
 4033 FORMAT(1X,5HOFUS=,F7.3,1X,6HNRTRI=,F7.3,1X,5HRHFO=,F7.3)          FCST0419
  206 IF(CONOPT(7) .EQ. 1 .AND. SDEPTH .GT. 0.0) WRITE(6,13)DATE,       FCST0420
     1SDEPTH,STMD,SAX,TANSM,SPLW                                        FCST0421
   13 FORMAT(3X,I4,2X,7HSDEPTH=,F8.2,2X,5HSTMD=,F6.2,2X,4HSAX=,F6.2,    FCST0422
     1  2X,6HTANSM=,F6.2,2X,5HSPLW=,F6.2)                               FCST0423
      MAXI=MAXI+1                                                       FCST0424
      MPDAY(MAXI)=DATE                                                  FCST0425
C   MONTHLY SUMMARY STORAGE                                            FCST0426
      STMROS(MAXI,3)=DSSF(DAY)                                          FCST0427
      TMSTF(MAXI,3)=AMSTF                                               FCST0428
      AMSTF = 0.0                                                       FCST0429
      TMRTF(MONTH) = AMRTF                                              FCST0430
      AMRTF = 0.0                                                       FCST0431
      EMBFNX(MAXI,3)=BFNX                                               FCST0432
      TMPREC(MAXI,3)=AMPREC                                             FCST0433
      AMPREC = 0.0                                                      FCST0434
      TMRPM(MONTH) = AMRPM                                              FCST0435
      AMRPM = 0.0                                                       FCST0436
      TMBF(MAXI,3)=AMBF                                                 FCST0437
      AMBF = 0.0                                                        FCST0438
      TMIF(MAXI,3)=AMIF                                                 FCST0439
      AMIF = 0.0                                                        FCST0440
      TMSE(MAXI,3)=AMSE                                                 FCST0441
      AMSE = 0.0                                                        FCST0442
      TMPET(MONTH) = AMPET                                              FCST0443
      AMPET = 0.0                                                       FCST0444
      TMNET(MONTH) = AMNET                                              FCST0445
      AMNET = 0.0                                                       FCST0446
      TMSNE(MONTH) = AMSNE                                              FCST0447
```

```
      AMSNE = 0.0                                                       FCST0448
      TMFSIL(MONTH) = AMFSIL                                            FCST0449
      AMFSIL = 0.0                                                      FCST0450
      EMGWS(MAXI,3)=GWS                                                 FCST0451
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZS/LZC)                        FCST0452
      IF(UZC .LT. 0.25) UZC = 0.25                                      FCST0453
      EMUZC(MAXI,3)=UZC                                                 FCST0454
      EMUZS(MAXI,3)=UZS                                                 FCST0455
      EMSIAM(MAXI,3)=SIAM                                               FCST0456
      EMLZS(MAXI,3)=LZS                                                 FCST0457
      EMIFS(MAXI,3)=IFS                                                 FCST0458
220   CONTINUE                                                          FCST0459
      IF  (MDAY.EQ.337) MDAY=59                                         FCST0460
C                                                                       FCST0461
C STORE MAXIMUM DAILY STREAM FLOW FOR YEAR                              FCST0462
      TFMAXY(MAXI)=TFMAX                                                FCST0463
3001  CONTINUE                                                          FCST0464
      IF (DAY.EQ.EDATE)  GO TO 3002                                     FCST0465
      CALL DAYNXT(DAY,DPY)                                              FCST0466
      IF (DAY.GT.LDAY) MDAY=LDAY                                        FCST0467
      IF (DAY.EQ.366)  MDAY=337                                         FCST0468
      GO TO 152                                                         FCST0469
3002  IF (CONOPT(16).EQ.0)  GO TO 221                                   FCST0470
      BACKSPACE 11                                                      FCST0471
      WRITE (11) (RPLOTC(I),I=1,1832)                                   FCST0472
      BACKSPACE 18                                                      FCST0473
      WRITE (18) (RCOMMA(I),I=1,12087)                                  FCST0474
C  END OF DAY LOOP                                                      FCST0475
  221 CONTINUE                                                          FCST0476
      RETURN                                                            FCST0477
      END                                                               FCST0478
```

```
      SUBROUTINE PASTRN(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)
      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMRDS(121,6),DPY,TITLE,
     2KFLAG,ICFLAG,
     1IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
     2  NCTRI,CTRI,FIRR,RICY,DPSE,BODFSM,SPBFIW,SPTWCC,SPM,ELCIF,
     3  XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPH,RGPMB,AREA,FIMP,
     A SATRI,UHFA,
     BMNRO,
     4  FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
     5  BIVF,OFSS,CFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,FXOPV,
     6  BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
     7  IFRL,LSHFT,NGTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,FPAET,FPER,
     8  TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPDR,UFUS,OFUSIS,OFP,OFRIS,PEIS,
     9  RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
     A  WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQOF,EQOFIS,SOFRF,SOFRFI,
     B  SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPIW,STMD,SFMD,ASMRG,
     XDEPEND(2),VARIN(2),MPTS,JULJI,IYR,TODARY(5,1),
     YTOMARY(7,1),TOSARY(5,6,1),TSMARY(6,1),TSMARY(8,1),
     ZTSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
     WTORARY(1,6,1),
     CORSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)
      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
     1TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
     2EPCM,SERA,SERR,SESF,SOER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
     3TMSTFI,T20OFH,T20PRH,TMRTFI,JULDAT,
     4TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
     5TRHF,
     5SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
C  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)
C     BASED ON STANFORD WATERSHED MODELS III & IV
      DIMENSION BTRI(99),CONOPT(15),CRFMI(22),CTRI(99),DDIW(366),
     1  DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
     2  DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
     3 EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
     4 EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
     5  RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SOER(22),
     6 THSF(24),TITLE(13),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
     7 TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
     8 TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T20OFH(21),T20PRH(21),
     . UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
     A PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)
      LOGICAL   LSHFT
      INTEGER   CDSDR,CN,CONOPT,DATE,DAY,DPY,EHSGD,HOUR,HRF,HRL,PDAY,
     1  PRD,RHPD,RHPH,RSBD,SGMD,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX
      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY
      INTEGER   DATES,EDATE,SINDEI,MPDAY(15)
      DIMENSION RPLOTC(1832),RCOMMA(12087)
      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))
      REAL  IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT
      DATA  MEDCY/  0,          31,59,90,120,151,181,212,243,273,304,334/
      DATA  MEDWY/304,334,365,31,59,90,120,151,181,212,243,273          /
      REAL      MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD
      REAL  SSQM,SSQDI,SSQMI,VDRSF,VDSSF
      REAL  VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
     *          SMSQM
      REAL  MDRSF,MDSSF,MMRSF,MMSSF
      BACKSPACE 11
      READ  (11) (RPLOTC(I),I=1,1832)
      BACKSPACE 18
      READ(18) (RCOMMA(I),I=1,12087)
      CALL READ(GWS,UZS,LZS,BFNX,IFS,UZC,SIAM)
      WRITE (6,5005) GWS,UZS,LZS,BFNX,IFS,UZC,SIAM
5005  FORMAT (1X,4HGWS=,F7.3,1X,4HUZS=,F7.3,1X,4HLZS=,F7.3,1X,5HBFNX=,
     1F7.3,1X,4HIFS=,F7.3,1X,4HUZC=,F7.3,1X,5HSIAM=,F7.3)
150   CONTINUE
      DATE=PDATE
      DAY=DATES
      MDAY=PDAY
      SINDEX=0
      MAXI=0
      AMRPM = 0.0
      AMPREC = 0.0
      AMBF = 0.0
      AMSE = 0.0
      AMSTF = 0.0
      AMRTF = 0.0
      IF(DPY .EQ. 366) MEDWY(5)=366
      WRITE  (6,4)  (TITLE(KTA),KTA=1,18)
4     FORMAT(1H1,10X,18A4,//)
3     FORMAT(1X,'PAST RUN HOURLY CFS VALUE',//,1X,4A4)
      WRITE(6,3)  MSBDIC
C  BEGIN DAY LOOP
      IMONTH=MONTH
  152 TDSF = 0.0
      IF (MONTH.NE.4) GO TO 148
      IF (MDAY.NE. 31) GO TO 149
148   IF (DATE.GT.(MOD(DAY,MDAY))) IMONTH=MONTH+1
149   CONTINUE
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      PET=EPCM(IMCNTH)*DPET(DAY)
      PETU = PET
      TFMAX = 0.0
C   EVAPOTRANSPIRATION ADJUSTMENTS
      IF(CONOPT(7) .NE. 1) GO TO 153
      IF(DMXT(DAY) - 4.0*ELDIF .LT. 40.0) PET = 0.0
      IF(SPTW .GT. SPTWCC) PET = FFOR*PET
C   CALCULATION OF SNOW EVAPORATION
      IF(DMNT(DAY) .GT. 32.0 .OR. SPTW .LE. DPSE(DAY)) GO TO 153
      SE = DPSE(DAY)
      AMSNE = AMSNE + SE
      SPTW = SPTW - SE
      IF(SFMD .GT. 0.0) SDEPTH = SDEPTH - SE/SFMD
  153 DO 202 HOUR = 1,24
      IF((NSGRD .EQ. 0) .AND. (DRHP(DAY,HOUR) .NE. 0.0)  .AND. (PET .EQ.
     1  PETU) .AND. (CONOPT(3) .EQ. 1)) PET = 0.5*PET
  154 IF(HOUR .EQ. SGRT + 1) RGPM = DRGPM(DAY)
      IF(HOUR .EQ. 9) HSE = (FWTR*PET)/12.0
      IF(HOUR .EQ.21) HSE = 0.0
      PRH = RGPM*DRHP(DAY,HOUR)
      AMPREC = AMPREC  + PRH
C   ENTER SNOWMELT SUBROUTINE
      IF(CCNUPT(7) .EQ. 1) CALL SNOMEL(BDDFSM,SPTWCC,SPM,ELDIF,DAY,
     1  SPBFLW, XDNFS,FFOR,FFSI,MRNSM,DSMGH,SDEPTH,STMD, PXCSA,HOUR,
     2  SAX,SOFRF,OFRFIS,SOFRFI,AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,
     3  WT4AM,WT4PM,ASM,ASMRG,SASFX,SARAX,DMXT,DMNT,RICY,FIRR)
  155 AMRPM = AMRPM + PRH
  156 TOFR = 0.0
      ARHF = 0.0
C  15 MINUTE ACCOUNTING AND ROUTING LOOP
      DO 187 PRD = 1,4
      PEBI = 0.0
      PPI = 0.0
      OFR = 0.0
      OFRIS = 0.0
      WI = 0.0
      WEIFS = 0.0
      PMEUZS = 0.0
      PMELZS = 0.0
      PMEIFS = 0.0
      PMEOFS = 0.0
      PEP = 0.25*PRH
      IF(CONOPT(2) .EQ. 1) CALL PREPRD(RGPM,DRHP,DAY,HOUR,DPY,PRD,PEP,
     1  PRH)
      IF(PEP .GT. 0.0) GO TO 157
      IF(OFUS .GT. 0.0) GO TO 159
      IF(IFS .GT. 0.0) GO TO 170
      IF(NRTRI .GT. 0) GO TO 172
      TRHF = 0.0
      IF(RHFO .GT. 0.0) GO TO 181
      GO TO 184
C   RAINFALL UPPER ZONE INTERACTION
  157 IF(PEP .GE. VINTCR) GO TO 158
      UZS = UZS + PEP*TPLR
      VINTCR = VINTCR - PEP
      PPI = 0.0
      PEBI = 0.0
      PMEUZS = PEP
      IF(OFUS .GT. 0.0) GO TO 159
      GO TO 170
  158 PPI = PEP - VINTCR
      UZS = UZS + VINTCR*TPLR
      VINTCR = 0.0
      LZSR = LZS/LZC
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZSR)
      IF(UZC .LT. 0.25) UZC = 0.25
      UZRX = 2.0*ABS(UZS/UZC - 1.0) + 1.0
      FMR = (1.0/(1.0 + UZRX))**UZRX
      IF(UZS .GT. UZC) FMR = 1.0 - FMR
      PEBI = PPI*FMR
      PMEUZS = PEP - PEBI
      UZS = UZS + PPI - PEBI
C   LOWER ZONE AND GROUNDWATER INFILTRATION
  159 LZSR = LZS/LZC
      EID = 4.0*LZSR
      IF(LZSR .LE. 1.0) GO TO 160
      EID = 4.0 + 2.0*(LZSR - 1.0)
      IF(LZSR .LE. 2.0) GO TO 160
      EID = 6.0
  160 PEBI = PEBI + OFUS
      CMIR = 0.25*SIAM*BMIR/(2.0**EID)
      CIVM = BIVF*2.0**LZSR
      IF(CIVM .LT. 1.0) CIVM = 1.0
      PEAI = PEBI*PEBI/(2.0*CMIR*CIVM)
      WI = PEBI*PEBI/(2.0*CMIR)
      IF(PEBI .GE. CMIR) WI = PEBI - 0.5*CMIR
      IF(PEBI .GE. CMIR*CIVM) PEAI = PERI - 0.5*CMIR*CIVM
      WEIFS = WI - PEAI
      IF(PEBI .LE. CFUS) GO TO 161
```

```
      PMELZS = (PEBI - WI)*((PEBI - OFUS)/PEBI)
      PMEIFS = WEIFS*((PEBI - OFUS)/PEBI)
      PMEOFS = PEAI*((PEBI - OFUS)/PEBI)
  161 CONTINUE
      IF((PEAI - OFUS) .GT. 0.0) GO TO 162
      EQD = (OFUS + PEAI)/2.0
      GO TO 163
  162 EQD = EQDF*((PEAI - OFUS)**0.6)
  163 IF((OFUS + PEAI) .GT. (2.0*EQD)) EQD = 0.5*(OFUS + PEAI)
      IF((OFUS + PEAI) .LE. 0.001) GO TO 164
      OFR = 0.25*OFRF*(((OFUS + PEAI)*0.5)**1.67)*((1.0 + 0.6*((OFUS +
     1  PEAI)/(2.0*EQD))**3.0)**1.67)
      IF(OFR .GT. (0.75*PEAI)) OFR = 0.75*PEAI
  164 IF(FIMP .EQ. 0.0) GO TO 168
  165 PEIS = PPI + OFUSIS
      IF((PEIS - OFUSIS) .GT. 0.0) GO TO 166
      EQDIS = (OFUSIS + PEIS)/2.0
      GO TO 167
  166 EQDIS = EQDFIS*((PEIS - OFUSIS)**0.6)
  167 IF((OFUSIS + PEIS) .GT. (2.0*EQDIS)) EQDIS = 0.5*(OFUSIS + PEIS)
      IF((OFUSIS + PEIS) .LE. 0.01) GO TO 168
      OFRIS = 0.25*OFRFIS*(((OFUSIS + PEIS)*0.5)**1.67)*((1.0 + 0.6*((
     1  OFUSIS + PEIS)/(2.0*EQDFIS))**3.0)**1.67)
      IF(OFRIS .GT. PEIS) OFRIS = PEIS
  168 TOFR = TOFR + FPER*OFR + FIMP*OFRIS + PPI*FWTR
      OFUSIS = PEIS - OFRIS
      OFUS = PEAI - OFR
      IF(OFUS .GE. 0.001) GO TO 169
      LZS = LZS + OFUS
      OFUS = 0.0
      OFRIS = OFRIS + OFUSIS
      OFUSIS = 0.0
  169 LZRX = 1.5*ABS(LZS/LZC - 1.0) + 1.0
      FMR = (1.0/(1.0 + LZRX))**LZRX
      IF(LZS .LT. LZC) FMR = 1.0 - FMR*(LZS/LZC)
      PLZS = FMR*(PEBI - WI)
      PGW = (1.0 -FMR)*(PEBI - WI)*(1.0 - SUBWF)*FPER
      GWS = GWS + PGW
      BFNX = BFNX + PGW
      LZS = LZS + PLZS
      IFS = IFS + WEIFS*FPER
  170 SPIF = IFRL*IFS
      AMIF = AMIF + SPIF
      IFS = IFS - SPIF
      IF(IFS .GE. 0.0001) GO TO 171
      LZS = LZS + IFS
      IFS = 0.0
  171 UHFA(1) = FPER*OFR + PPI*FWTR + FIMP*OFRIS + SPIF
      SPDR = UHFA(1)
C  ROUTING
  172 IF(CONOPT(12) .NE. 1) GO TO 173
      URHF = URHF + 0.25*UHFA(1)
      IF(PRD .NE. 4) GO TO 181
      UHFA(1) = URHF
  173 TRHF = 0.0
      KTRI = NCTRI
      IF(CONOPT(13) .EQ. 1) KTRI = NCSTRI
  174 URHF = UHFA(KTRI)
      IF(URHF .LE. 0.0) GO TO 176
  175 TRHF = TRHF + URHF*CTRI(KTRI)
      IF(CONOPT(13) .EQ. 1 .AND. LSHFT .AND. KTRI .GE. 2) TRHF = TRHF +
     1  URHF*SATRI(KTRI - 1)
      UHFA(KTRI + 1) = URHF
      GO TO 177
  176 UHFA(KTRI+ 1) = 0.0
  177 KTRI = KTRI - 1
      IF(KTRI .GE. 1) GO TO 174
  178 IF(URHF .LE. 0.0) GO TO 179
      NRTRI = NCTRI
      IF(CONOPT(13) .EQ. 1) NRTRI = MXTRI
  179 NRTRI = NRTRI - 1
      UHFA(1) = 0.0
      IF(CONOPT(13) .NE. 1) GO TO 180
      NNSTRI = NCSTRI + 1
      UHFA(NNSTRI) = 0.0
  180 URHF = 0.0
  181 IF(SRX .LE. CSRX) SRX = CSRX
      RHF1 = TRHF - SRX*(TRHF - RHFO)
      RHFO = RHF1
      IF(RHFO .LT. RHFMC) RHFO = 0.0
      TFCFS = (4.0*RHF1 + CBF - HSF)*WCFS
      IF(CONOPT(13) .NE. 1) GO TO 182
      IF(CONOPT(12) .EQ. 1 .AND. PRD .NE. 4) GO TO 182
      CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
     1  TFCFS)
      DATE = MOD(DAY,MDAY)
      IF(LSHFT) WRITE(6,6) DATE,HOUR,PRD,NCSTRI
    6 FORMAT(2X,I2,2X,I2,2X,I2,2X,20HHISTOGRAM CHANGES TO,1X,I2,1X,
     1  8HELEMENTS)
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
  182 CONTINUE
      IF(TFCFS .LE. 0.5*CHCAP) SRX = CSRX
      IF((TFCFS .GT. 0.5*CHCAP) .AND. (TFCFS .LT. 2.0*CHCAP)) SRX = CSRX
     1 +(FSRX - CSRX)*((TFCFS - 0.5*CHCAP)/(1.5*CHCAP))**3
      IF(TFCFS .GT. 2.0*CHCAP) SRX = FSRX
      IF(TFCFS .LE. TFMAX) GO TO 183
      PRDF = PRD
      TDFP24 = HOUR
      IF(PRD .LE. 3) TDFP24 = (TDFP24 - 1.0) + 0.15*PRDF
      TFMAX = TFCFS
  183 ARHF = ARHF + RHF1
C   STORM OUTPUT REQUESTED BY CONOPT(1)
  184 IF(CONOPT(1) .NE. 1) GO TO 186
      IF(DAY .NE. CDSDR) GO TO 186
      IF(HOUR .EQ. 1 .AND. PRD .EQ. 1) WRITE(6,7)
    7 FORMAT(1H//,21X,19HRAINFALL DEPOSITION,12X,16HMOISTURE STORAGE,
     1  14X,17HSTREAMFLOW ORIGIN,6X,14HSTREAM OUTFLOW/2X,116HDY HR PD RA
     2IN   EUZS    ELZS    EIFS    EOFS          UZS     LZS     IFS      OFS     S
     3POF    SPIF    SPBF    SPTF  INCHES      CFS)
      DATE = MOD(DAY,MDAY)
      OFS = OFUS*FPER + OFUSIS*FIMP
      SPOF = OFR*FPER + OFRIS*FIMP + PPI*FWTR
      SPBF = 0.25*(CBF-HSE)
      SPTF = SPDR + SPBF
      SPDR = 0.0
      IF(RHFO .LE. 0.0) TFCFS = (CBF - HSE)*WCFS
      RSPTF = 0.25*TFCFS/WCFS
      WRITE(6,8) DATE,HOUR,PRD,PEP,PMEUZS,PMELZS,PMEIFS,PMEOFS,UZS,LZS
     1,IFS,OFS,SPOF,SPIF,SPBF,SPTF,RSPTF,TFCFS
    8 FORMAT(2X,I2,1X,I2,1X,I1,5(1X,F6.4),2X,4(F7.4),2X,5(1X,F6.4),1X,
     1  F7.1)
      IF(HOUR .EQ. 24 .AND. PRD .EQ. 4) GO TO 185
      GO TO 186
  185 NDSDP = NDSDP + 1
      IF(NDSDR .EQ. NDSDP) GO TO 186
      CALL DAYNXT(CDSDR,DPY)
  186 CONTINUE
      IF(VINTCR .LT. 0.25*VINTMR) VINTCR = VINTCR + DPET(DAY)/96.0
  187 CONTINUE
C   END OF 15 MINUTE LOOP
      IF(CONOPT(5) .NE. 1) GO TO 197
C   HOURLY OVERLAND FLOW AND RAINFALL SORTING
      IF(TOFR .LE. 0.0) GO TO 193
      KT20 = 20
  188 IF(KT20 .LT. 1) GO TO 192
      IF(TOFR .GT. T20OFH(KT20)) GO TO 189
      GO TO 190
  189 T20OFH(KT20+1) = T20OFH(KT20)
      GO TO 191
  190 T20OFH(KT20+1) = TOFR
      GO TO 193
  191 KT20 = KT20 - 1
      GO TO 188
  192 T20OFH(1) = TOFR
  193 IF(PRH .LE. 0.0) GO TO 197
      KT20 = 20
  194 IF(KT20 .LT. 1) GO TO 196
      IF(PRH .GT. T20PRH(KT20)) GO TO 195
      T20PRH(KT20 + 1) = PRH
      GO TO 197
  195 T20PRH(KT20+1) = T20PRH(KT20)
      KT20 = KT20 - 1
      GO TO 194
  196 T20PRH(1) = PRH
C   ADDING GROUNDWATER FLOW
  197 CBF = GWS*BFRL*(1.0 + BFNRL*BFNX)
      GWS = GWS - CBF
      AMBF = AMBF + CBF
      THGR = ARHF + CBF
      IF(HSE .GT. THGR) HSE = THGR
      AMSE = AMSE + HSE
      THSF(HOUR) = (THGR - HSE)*WCFS
      TOSF = TOSF + THSF(HOUR)
C
C STORE SIMULATED HOURLY STREAM FLOWS
C   DRAINING OF UPPER ZONE STORAGE
      SINDEX=SINDEX+1
      THSFD(SINDEX,1)=THSF(HOUR)
306   UZINFX=(UZS/UZC) - (LZS/LZC)
      IF(UZINFX .LE. 0.0) GO TO 198
      LZSR = LZS/LZC
      UZINLZ = 0.003*BMIR*UZC*UZINFX**3.0
      IF(UZINLZ .GT. UZS) UZINLZ = UZS
      UZS = UZS - UZINLZ
      LZRX = 1.5*ABS(LZSR - 1.0) + 1.0
      FMR = (1.0/(1.0 + LZRX))**LZRX
      IF(LZS .LT. LZC) FMR = 1.0 - FMR*LZSR
      PGW = (1.0-FMR)*UZINLZ*(1.0 - SUBWF)*FPER
      PLZS = FMR*UZINLZ
```

```
      LZS = LZS + PLZS                                                  PAST0354
      GWS = GWS + PGW                                                   PAST0355
      BFNX = BFNX + PGW                                                 PAST0356
C   4 PM ADJUSTMENTS OF VARIOUS VALUES                                  PAST0357
  198 IF(HOUR .NE. 16) GO TO 202                                        PAST0358
      AEX90 = 0.9*(AEX90 + PET)                                         PAST0359
      AEX96 = 0.96*(AEX96 + PET)                                        PAST0360
C   INFILTRATION CORRECTION                                             PAST0361
      SIAM = (AEX96/AETX)**SIAC                                         PAST0362
      IF(SIAM .LT. 0.33) SIAM = 0.33                                    PAST0363
      BFNX = 0.97*BFNX                                                  PAST0364
      IF(PET .EQ. 0.0) GO TO 202                                        PAST0365
C   EVAP-TRANS LOSS FROM GROUNDWATER                                    PAST0366
      GWET = GWS*GWETF*PET*FPER                                         PAST0367
      GWS = GWS - GWET                                                  PAST0368
      BFNX = BFNX - GWET                                                PAST0369
      IF(BFNX .LT. 0.0) BFNX = 0.0                                      PAST0370
      AMPET = AMPET + PET                                               PAST0371
      IF(PET .GE. UZS) GO TO 199                                        PAST0372
      UZS = UZS - PET                                                   PAST0373
      AMNET = AMNET + PET                                               PAST0374
      GO TO 202                                                         PAST0375
  199 PET = PET - UZS                                                   PAST0376
      AMNET = AMNET + UZS                                               PAST0377
      UZS = 0.0                                                         PAST0378
      LZSR = LZS/LZC                                                    PAST0379
      IF(PET .GE. ETLF*LZSR) GO TO 200                                  PAST0380
      SET = PET*(1.0 - PET/(2.0*ETLF*LZSR))                             PAST0381
      GO TO 201                                                         PAST0382
  200 SET = 0.5*ETLF*LZSR                                               PAST0383
  201 LZS = LZS - SET                                                   PAST0384
      AMNET = AMNET + SET                                               PAST0385
  202 CONTINUE                                                          PAST0386
C   END OF HOUR LOOP                                                    PAST0387
      DSSF(DAY) = TDSF/24.0                                             PAST0388
      IF(CONOPT(11) .EQ. 1) DSSF(DAY) = DSSF(DAY) + DDIW(DAY)           PAST0389
  203 AMRTF = AMRTF + DRSF(DAY)                                         PAST0390
      AMSTF = AMSTF + DSSF(DAY)                                         PAST0391
      IF(CONOPT(6) .EQ. 1) EDLZS(DAY) = LZS                             PAST0392
C   STORE ERRORS AND FLOW DURATION                                      PAST0393
      IF(CONOPT(4) .NE. 1) GO TO 204                                    PAST0394
      ERR = DSSF(DAY) - DRSF(DAY)                                       PAST0395
      IF(DRSF(DAY) .LT. 1.0) KRFMI = 1.0                                PAST0396
      IF(DRSF(DAY) .GT. 1.0) KRFMI= 2.0*ALOG(DRSF(DAY)) + 2.0           PAST0397
      CRFMI(KRFMI) = CRFMI(KRFMI) + 1.0                                 PAST0398
      SERR(KRFMI) = SERR(KRFMI) + ERR                                   PAST0399
      SERA(KRFMI) = SERA(KRFMI) + ABS(ERR)                              PAST0400
      SQER(KRFMI) = SQER(KRFMI) + ERR*ERR                               PAST0401
      SESF(KRFMI) = 0.0                                                 PAST0402
      IF(CRFMI(KRFMI) .GT. 1.0) SESF(KRFMI) = SQRT(ABS((SQER(KRFMI) -   PAST0403
     1  SERR(KRFMI)**2/CRFMI(KRFMI))/(CRFMI(KRFMI) - 1.0)))             PAST0404
204   CONTINUE                                                          PAST0405
      DATE=DAY                                                          PAST0406
      IF ((MONTH.EQ.4).AND.(MDAY.EQ.31)) DATE=MOD(DAY,MDAY)             PAST0407
      IF (MONTH.NE.4) DATE=MOD(DAY,MDAY)                                PAST0408
      WRITE(6,9) DATE, (THSF(HOUR),HOUR=1,12)                           PAST0409
    9 FORMAT(1H/,1X/,1X,I4,2X,2HAM,1X,6F8.1,3X,6F8.1)                   PAST0410
      WRITE(6,10) (THSF(HOUR),HOUR=13,24), DSSF(DAY)                    PAST0411
   10 FORMAT(1HJ,6X,2HPM,1X,6F8.1,3X,7F8.1)                             PAST0412
      IF(TDFP24 .LT. 12.0) GO TO 205                                    PAST0413
      TDFP12 = TDFP24 - 12.0                                            PAST0414
   11 FORMAT(1H/,10X,8HMAXIMUM=,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,  PAST0416
     1  4HP.M.)                                                         PAST0417
      GO TO 206                                                         PAST0418
205   CONTINUE                                                          PAST0000
   12 FORMAT(1H/,10X,8HMAXIMUM=,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,  PAST0420
     1  4HA.M.)                                                         PAST0421
  206 IF(CONOPT(7) .EQ. 1 .AND. SDEPTH .GT. 0.0) WRITE(6,13)DATE,       PAST0422
     1SDEPTH,STMD,SAX,TANSM,SPLW                                        PAST0423
   13 FORMAT(3X,I4,2X,7HSDEPTH=,F8.2,2X,5HSTMD=,F6.2,2X,4HSAX=,F6.2,    PAST0424
     1  2X,6HTANSM=,F6.2,2X,5HSPLW=,F6.2)                               PAST0425
      MAXI=MAXI+1                                                       PAST0426
      MPDAY(MAXI)=DATE                                                  PAST0427
C   MONTHLY SUMMARY STORAGE                                             PAST0428
      STMROS(MAXI,1)=DSSF(DAY)                                          PAST0429
      TMSTF(MAXI,1)=AMSTF                                               PAST0430
      AMSTF = 0.0                                                       PAST0431
      TMRTF(MONTH) = AMRTF                                              PAST0432
      AMRTF = 0.0                                                       PAST0433
      EMBFNX(MAXI,1)=BFNX                                               PAST0434
      TMPREC(MAXI,1)=AMPREC                                             PAST0435
      AMPREC = 0.0                                                      PAST0436
      TMRPM(MONTH) = AMRPM                                              PAST0437
      AMRPM = 0.0                                                       PAST0438
      TMBF(MAXI,1)=AMBF                                                 PAST0439
      AMBF = 0.0                                                        PAST0440
      TMIF(MAXI,1)=AMIF                                                 PAST0441
      AMIF = 0.0                                                        PAST0442
      TMSE(MAXI,1)=AMSE                                                 PAST0443
```

```
      AMSE = 0.0
      TMPET(MONTH) = AMPET
      AMPET = 0.0
      TMNET(MONTH) = AMNET
      AMNET = 0.0
      TMSNE(MONTH) = AMSNE
      AMSNE = 0.0
      TMFSIL(MONTH) = AMFSIL
      AMFSIL = 0.0
      EMGWS(MAXI,1)=GWS
      UZC = SUZC*AEX90 + BUZC*EXP(-2.7*LZS/LZC)
      IF(UZC .LT. 0.25) UZC = 0.25
      EMUZC(MAXI,1)=UZC
      EMUZS(MAXI,1)=UZS
      EMSIAM(MAXI,1)=SIAM
      EMLZS(MAXI,1)=LZS
      EMIFS(MAXI,1)=IFS
220   CONTINUE
      IF  (MDAY.EQ.337) MDAY=59
C
C STORE MAXIMUM DAILY STREAM FLOW FOR YEAR
      TFMAXY(MAXI)=TFMAX
3001  CONTINUE
      IF (DAY.EQ.EDATE) GO TO 3002
      CALL DAYNXT(DAY,DPY)
      IF (DAY.GT.LDAY) MDAY=LDAY
      IF (DAY.EQ.366)  MDAY=337
      GO TO 152
3002  IF(CONOPT(16).EQ.0) GO TO 221
      BACKSPACE 11
      WRITE (11) (RPLOTC(I),I=1,1832)
      BACKSPACE 18
      WRITE (18) (RCOMMA(I),I=1,12087)
C   END OF DAY LOOP
  221 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE OUTPUT(DIFFS,DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,
     1                  APREC,DIFFPR,DIFPP,QOUT,QMAX,PHRO,MSBDIC,
     2                  MPDAY)
      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,SIMRUS(121,6),DPY,TITLF,
     2KFLAG,ICFLAG,
     1IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
     2  NCTRI,CTRI,FIRR,RICY,DPSE,BDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,
     3  XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
     A SATRI,UHFA,
     BMNRD,
     4  FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
     5  BIVF,OFSS,OFSL,OFMN,OFMNIS,IFFC,CSRX,FSRX,CHCAP,EXQPV,
     6  BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHF,IFPRC,
     7  IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAFT,FPFR,
     8  TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPOR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
     9  RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
     A  WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQFFIS,SUFRF,SCFRFI,
     B  SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
     XDEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
     YTOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
     ZTSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
     WTORARY(1,6,1),
     CDRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)
      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMOF,
     1TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
     2EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNF,
     3TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,
     4TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
     5TRHF,
     5SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
C  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)
C     BASED ON STANFORD WATERSHED MODELS III & IV
      DIMENSION BTRI(99),CONOPT(16),CRFMI(22),CTRI(99),DDIW(366),
     1  DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
     2  DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
     3 EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
     4 EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
     5  RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),
     6 THSF(24),TITLF(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
     7 TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSF(15,3),
     8 TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T200FH(21),T20PRH(21),
     . UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
     A PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)
      REAL*8 MSVT(9''MOISTURE',' STORAGE',' VALUES ',
     1'        T','ABLE 3A ',4*'          '/
      REAL*8 SUBT(11)/'ABLE 3A','ABLE 3B','    PAST',' 3A     ','UES
     1 ','PAST RUN','ABLE 1A ','ABLE 1B ',' 3B      ','MMARY    ',
     1'   PAST '/
      REAL QOUT(15),SUM(3),APREC(3),MSBDIC
      INTEGER MPDAY(15),TYPE,CONOPT
      REAL*4 SCPAST(36)/18*'    ',3*'    ','OBSE','RVED','    ',
     1'SIMU','LATE','D   ',3*'    ','OBSE','RVED','    ','SIMU','LATE',
     2'D   '/
      REAL*4 MSPAST(36)/20*'    ','SIMU','LATE','D   ',5*'    ','SIMU',
     1'LATE','D   ',5*'    '/
      REAL*4 SSPAST(36)/21*'    ','   O','BSER','VED ','   S','IMUL',
     1'ATED','    ','DIFF','    ','%DIF','F   ',4*'    '/
      REAL*8 STST(9)/'STORM SU','MMARY TA','BLE 2   ',6*'        '/
      REAL*8      STRMT(9)/'STORM CH','ARACTERI','STICS  T','ABLE 1A ',
     15*'        '/
      REAL*8      STCHAR(252)/'        ','STORM CH','ARACTERI','STICS   '
     1,'FORECAST',' RUN   T','ABLE 1A ',12*'        ','    WORS',
     2'T   NO  ','   F/CA','ST      ','        ','WORST   ','NO      ',
     3'F/CAST  ','        ','    CASE','    PREC','IP      ',
     42*'        ','CASE    ','PRECIP  ',9*'        ','        ',
     8'    PREC',
     5'IP      ',3*'        ','PRECIP  ',3*'        ','     SUR ',
     6197*'        '/
      REAL*8 STCSP(197)/'R/O     ',
     63*'        ','SUR R/O ',3*'        ','     INT ',
     7'FL      ',3*'        ','INT FL  ',3*'        ','    BASE',
     8' FL     ',3*'        ','BASE FL ',3*'        ','     STM ',
     9'R/O     ',3*'        ','STM R/O ',3*'        ',9*'        ',
     APREC','IP      ',3*'        ','PRECIP  ',3*'        ','     SUR ',
     B'R/O     ',3*'        ','SUR R/O ',3*'        ','     INT ','FL
     C ',3*'        ','INT FL  ',3*'        ','    BASE',' FL     ',3*'
     D        ','BASE FL ',3*'        ','     STM ','R/O     ',3*'
     E ','STM R/O ',3*'        ',9*'        ','    PREC','IP      ',3*'
     F        ','PRECIP  ',3*'        ','     SUR ','R/O     ',3*'
     S','SUR R/O ',3*'        ',
     G '     INT ','FL      ',3*'        ','INT FL  ',3*'        ','
     HBASE',' FL     ',3*'        ','BASE FL ',3*'        ','     STM ','
     IR/O     ',3*'        ','STM R/O ',3*'        ',9*'        ','
     JREC','IP      ',3*'        ','PRECIP  ',3*'        ','     SUR ',
     K'R/O     ',3*'        ','SUR R/O ',3*'        ','     INT ','FL
     L ',3*'        ','INT FL  ',3*'        ','    BASE',' FL     ',3*'
     K        ','BASE FL ',3*'        ','     STM ','R/O     ',3*'
     M ','STM R/O ',3*'        '/
      REAL*8 STSURD(208)     /2*'        ','STORM SU','MMARY  F','ORECAS
```

```
      AT ','RUN  TAB','LE 2    ',13*'          ','WORST   ','NO      ','F/C
     BAST   ','DIFF     ','%DIFF    ',4*'      ','CASE     ','PRECIP  ',14
     C*'         ','PEAK    (','CFS)       ',7*'           ',9*'      ',
     D'PEAK    (','HR)          ',16*'         ','R/O     (','IN)       ',16*'
     E       ','PRECIP (','IN)           ',7*'          '/
      REAL*8 MSVARD(441)/'           ','         MOIS','TURE STO','RAGE VAL',
     1'UES FORE','CAST RUN','   TABLE',' 3A      ',10*'             ',
     2'          ','WURST     ','NO          ','F/CAST   ','            ',
     3'WORST    ','NO        ','F/CAST    ',2*'          ','CASE          ',
     4'PRECIP   ',2*'             ','CASE        ','PRECIP  ',2*'           ',
     59*'          ','        IFS  ',3*'           ','        IFS  ',4*'       ',
     6'      UZS  ',3*'       ','     UZS  ',4*'       ',
     7'      LZS  ',3*'       ','     LZS  ',4*'       ',
     8'      GWS  ',3*'       ','     GWS  ',4*'       ',
     9'      BFNX',3*'        ','     BFNX',4*'        ',
     A'      SIAM',3*'        ','     SIAM',4*'        ',
     B'      UZC  ',3*'       ','     UZC  ',4*'       ',
     89*'          ',
     1'      IFS  ',3*'       ','     IFS  ',4*'       ',
     2'      UZS  ',3*'       ','     UZS  ',4*'       ',
     2'      LZS  ',3*'       ','     LZS  ',4*'       ',
     4'      GWS  ',3*'       ','     GWS  ',4*'       ',
     5'      BFNX',3*'        ','     BFNX',4*'        ',
     6'      SIAM',3*'        ','     SIAM',4*'        ',
     7'      UZC  ',3*'       ','     UZC  ',4*'       ',261*'       '/
      REAL*8 MSVSUD(135)/
     1'      IFS  ',3*'       ','     IFS  ',4*'       ',
     2'      UZS  ',3*'       ','     UZS  ',4*'       ',
     2'      LZS  ',3*'       ','     LZS  ',4*'       ',
     4'      GWS  ',3*'       ','     GWS  ',4*'       ',
     5'      BFNX',3*'        ','     BFNX',4*'        ',
     6'      SIAM',3*'        ','     SIAM',4*'        ',
     7'      UZC  ',3*'       ','     UZC  ',4*'       ',
     89*'          ',
     1'      IFS  ',3*'       ','     IFS  ',4*'       ',
     2'      UZS  ',3*'       ','     UZS  ',4*'       ',
     2'      LZS  ',3*'       ','     LZS  ',4*'       ',
     4'      GWS  ',3*'       ','     GWS  ',4*'       ',
     5'      BFNX',3*'        ','     BFNX',4*'        ',
     6'      SIAM',3*'        ','     SIAM',4*'        ',
     7'      UZC  ',3*'       ','     UZC  ',4*'       '/
      REAL      BLANK/'    '/
      REAL*4  STCHAS(648)
      REAL*4  STCSS(394)
      REAL*4  MSVARY(648)
      REAL*4  STSMRY(416)
      REAL*4  MSVSUP(270)
      EQUIVALENCE  (MSVARD(1),MSVARY(1))
      EQUIVALENCE  (MSVSUD(1),MSVSUP(1))
      EQUIVALENCE  (STSURD(1),STSMRY(1))
      EQUIVALENCE(STCHAR(1),STCHAS(1))
      EQUIVALENCE  (STCSP(1),STCSS(1))
      TYPE=4
      MLN=72
      J=0
      MI=3
      NOM=28
      DO  23  K=1,394
      STCHAS(110+K)=STCSS(K)
   23 CONTINUE
      IF(CONOPT(15).NE.2)  GO TO 61
      J=3
      MI=1
      DO  65  I=1,36
      STCHAS(36+I)=SCPAST(I)
   65 CONTINUE
      STCHAR(5)=SUBT(3)
   61 MAX=MAXI
      STCHAS(55)=MSBDIC
      LI=1
   13 CONTINUE
      K=0
      L=0
      DO  10  I=1,MAX
      IR=1
      NUM=2
      CALL  CONVER(MPDAY(LI),STCHAS(91+K+L),IR,NUM)
      DO  11  M=1,MI
      NUM=3
      IR=0
      CALL  CCNVFR(TMPREC(I,M),STCHAS((92+(2*M))+K+L),IR,NUM)
      IF  (CONOPT(15).EQ.2)  CALL  CONVEP(TMPREC(I,1),STCHAS(97+K+L),IR,
     1NUM)
      CALL  CCNVER(TMOF(I,M),STCHAS((110+(2*M))+J+K+L),IR,NUM)
      CALL  CCNVER(TMIF(I,M),STCHAS((128+(2*M))+J+K+L),IR,NUM)
      CALL  CCNVER(TMBF(I,M),STCHAS((146+(2*M))+J+K+L),IR,NUM)
      CALL  CCNVER(STMROS(I,M),STCHAS((164+(2*M))+K+L),IR,NUM)
      IF  (CONCPT(15).EQ.2)  CALL  CONVER(QOUT(I),STCHAS(166+K+L),IR,NUM)
      IF  (CONUPT(15).EQ.2)  CALL  CONVER(STMROS(I,1),STCHAS(169+K+L),IR,
     1NUM)
```

```
11    CONTINUE
      K=K+108
      LI=LI+1
      IF(MOD(I,4).EQ.0) L=9
      IF(MOD(I,4).EQ.0) K=0
      IF(MOD(I,8).EQ.0) GO TO 17
      IF (I.EQ.MAX)  GO TO 17
      GO TO 10
17    CALL DVAR(TYPE,STCHAS,MLN,NOM,STRMT,LVAL,HVAL,VDL,TDATA,NCODE)
      STCHAR(7)=SUBT(9)
      STRMT(4)=SUBT(8)
      IF (I.EQ.MAX) GO TO 10
      DO 15 N=1,325,108
      STCHAS(90+N)=BLANK
      STCHAS(99+N)=BLANK
      DO 18  K=1,6
      STCHAS(92+N+K)=BLANK
      STCHAS(101+N+K)=BLANK
      STCHAS(110+N+K)=BLANK
      STCHAS(119+N+K)=BLANK
      STCHAS(128+N+K)=BLANK
      STCHAS(137+N+K)=BLANK
      STCHAS(146+N+K)=BLANK
      STCHAS(155+N+K)=BLANK
      STCHAS(164+N+K)=BLANK
      STCHAS(173+N+K)=BLANK
18    CONTINUE
15    CONTINUE
      K=0
      L=0
10    CONTINUE
      IF (CONOPT(15).NE.2) GO TO 32
      STSURD(5)=SUBT(3)
      STSURD(4)=SUBT(11)
      DO 39 I=1,36
      STSMRY(36+I)=SSPAST(I)
39    CONTINUE
      IR=0
      NUM=4
      CALL CONVER(QMAX,STSMRY(95),IR,NUM)
      CALL CONVER(PEAKS(1),STSMRY(98),IR,NUM)
      CALL CONVER(APREC(1),STSMRY(203),IR,NUM)
      CALL CONVER(APREC(1),STSMRY(207),IR,NUM)
      IR=1
      NUM=3
      CALL CONVER(PHRO,STSMRY(131),IR,NUM)
      CALL CONVER(PHRS(1),STSMRY(134),IR,NUM)
      IR=0
      CALL CONVER(SUM(1),STSMRY(167),IR,NUM)
      CALL CONVER(SUM(2),STSMRY(170),IR,NUM)
      GO TO 35
32    DO 31 I=1,3
      IR=0
      NUM=4
      CALL CONVER(PEAKS(I),STSMRY(93+(2*I)),IR,NUM)
      CALL CONVER(APREC(I),STSMRY(201+(2*I)),IR,NUM)
      NUM=3
      CALL CONVER (SUM(I),STSMRY(165+(2*I)),IR,NUM)
      IR=1
      CALL CONVER(PHRS(I),STSMRY(129+(2*I)),IR,NUM)
31    CONTINUE
35    IR=0
      NUM=4
      NOM=12
      CALL CONVER(DIFFS,STSMRY(101),IR,NUM)
      CALL CONVER(DIFFP,STSMRY(137),IR,NUM)
      CALL CONVER(DIFPP,STSMRY(211),IR,NUM)
      IF (CONOPT(15).EQ.2) STSMRY(211)=BLANK
      CALL CONVER(DIFFMP,STSMRY(103),IR,NUM)
      CALL CONVER(DIFFPP,STSMRY(139),IR,NUM)
      CALL CONVER(DIFFR,STSMRY(173),IR,NUM)
      CALL CONVER(DIFFRP,STSMRY(175),IR,NUM)
      CALL CONVER(DIFFPR,STSMRY(209),IR,NUM)
      IF (CONOPT(15).EQ.2) STSMRY(209)=BLANK
      CALL DVAR(TYPE,STSMRY,MLN,NOM,STST,LVAL,HVAL,VDL,TDATA,NCODE)
      DO 49 I=1,270
      MSVARY(378+I)=MSVSUP(I)
49    CONTINUE
      MI=3
      IF (CONOPT(15).NE.2) GO TO 63
      DO 81 I=1,36
      MSVARY(36+I)=MSPAST(I)
81    CONTINUE
      MSVARD(5)=SUBT(5)
      MSVARD(6)=SUBT(6)
      MI=1
63    NOM=40
      L=0
      L1=1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      K=0
      MAX=MAXI
      MSVARY(55)=MSBDIC
      DO 41 I=1,MAX
      IR=1
      NUM=2
      CALL CONVER(MPDAY(L1),MSVARY(91+K+L),IR,NUM)
      DO 43 M=1,MI
      IR=0
      NUM=3
      CALL CONVER(EMIFS(I,M),MSVARY(91+(2*M)+K+L),IR,NUM)
      CALL CONVER(EMUZS(I,M),MSVARY(109+(2*M)+K+L),IR,NUM)
      CALL CONVER(EMLZS(I,M),MSVARY(127+(2*M)+K+L),IR,NUM)
      CALL CONVER(EMGWS(I,M),MSVARY(145+(2*M)+K+L),IR,NUM)
      CALL CONVER(EMUZC(I,M),MSVARY(199+(2*M)+K+L),IR,NUM)
      CALL CONVER(EMSIAM(I,M),MSVARY(181+(2*M)+K+L),IR,NUM)
      CALL CONVER(EMBFNX(I,M),MSVARY(163+(2*M)+K+L),IR,NUM)
   43 CONTINUE
      K=K+144
      L1=L1+1
      IF (MOD(I,4).EQ.0) K=0
      IF (MOD(I,4).EQ.0) L=8
      IF (MOD(I,8).EQ.0) GO TO 45
      IF (I.EQ.MAX) GO TO 45
      GO TO 41
   45 CALL DVAR(TYPE,MSVARY,MLN,NOM,MSVT,LVAL,HVAL,VL,TDATA,NCODE)
      IF (I.EQ.MAX) GO TO 41
      MSVT(5)=SUBT(2)
      MSVARD(8)=SUBT(9)
      DO 53 N=1,434,144
      MSVARY(90+N)=BLANK
      MSVARY(98+N)=BLANK
      DO 55 K=1,7
      MSVARY(91+N+K)=BLANK
      MSVARY(99+N+K)=BLANK
      MSVARY(109+N+K)=BLANK
      MSVARY(117+N+K)=BLANK
      MSVARY(127+N+K)=BLANK
      MSVARY(135+N+K)=BLANK
      MSVARY(145+N+K)=BLANK
      MSVARY(153+N+K)=BLANK
      MSVARY(163+N+K)=BLANK
      MSVARY(171+N+K)=BLANK
      MSVARY(181+N+K)=BLANK
      MSVARY(189+N+K)=BLANK
      MSVARY(199+N+K)=BLANK
      MSVARY(207+N+K)=BLANK
   55 CONTINUE
   53 CONTINUE
      K=0
      L=0
   41 CONTINUE
      CALL PLTEND(0)
      RETURN
      END
```

```
      SUBROUTINE ATTEN
C     THIS IS THE 2250 OPERATOR INTERFACE ROUTINE
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     1       NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2       IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3       VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4       XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5       YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6       AY     ,BY     ,YLPOS  ,
     7       LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8       FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      LOGICAL*1 GSPIN, DUM(2)
      REAL*8 WORK(36)
      INTEGER*4 INTAR(10),GSP1,UNITN,NULLV(1),OAREA
      LOGICAL*1 FA,FB,FC,GMODE1,GSPIN
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      REAL*4 RNTAR(10),XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      EQUIVALENCE(INTAR(1),RNTAR(1),WORK(1)),(ISYM,ICODE)
      ND3 = IABS(IDSP)+1
      CALL ENATN(IATN,0,28,34)
    5 CONTINUE
      CALL INCL(IGDS5,NULLV,IKEY1)
C     REQUEST ATTENTION INFORMATION
   10 CALL RQATN(IATN,ICODE,2,INTAR,0,28,34)
      IF(ICODE)20,10,30
   20 CALL TMGSP(GSP1)
      WRITE(NOUT,1000)
      STOP
   22 RETURN
C
   30 CONTINUE
      IF(ICODE-28)35,32,35
   32 GMODE1=.NOT.GMODE1
      WRITE(NOUT,1020)GMODE1
      GO TO 10
   35 CONTINUE
      IF(INTAR(1) .EQ. IGDS6)GO TO 200
      IF(INTAR(1) .NE. IGDS5)GO TO 10
      IF(INTAR(2) .EQ. IKEY1)GO TO 40
      IF(INTAR(2) .EQ. IKEY2)GO TO 400
      IF(INTAR(2) .EQ. IKEY3)GO TO 100
C
   40 CONTINUE
      CALL DSATN(IATN,0,28,34)
      CALL OMIT(IGDS5,NULLV,IKEY1)
      RETURN
C
C     BRING UP PLOT DICTIONARY.  THE DICTIONARY CAN ONLY BE REQUESTED
C     FROM THE LATEST DISPLAY
C
  100 CONTINUE
      CALL OMIT(IGDS5)
      CALL OMIT(IGDSC)
  103 CALL INCL(IGDS6)
      ND2=MIN0(ND3,1000)
      ND1=1
      IF (ND3-1000) 105,105,104
  104 CALL INCL(IGDS6)
      CALL INCL(IGDS6,-3,NULLV)
  105 CALL INCL(IGDS6,-2,NULLV)
      CALL INCL(IGDS6,-1,NULLV)
      DO 110 I=ND1,ND2
  110 CALL INCL(IGDS6,I,NULLV)
      GO TO 10
C
C     PLOT DICTIONARY ATTENTION ROUTINE
C
  200 CONTINUE
      IF(INTAR(4))205,205,300
  205 IF(INTAR(4) .EQ. -1)GO TO 10
      IF(INTAR(4) .EQ. -3)GO TO 250
      IF(INTAR(4) .NE. -2)GO TO 10
  210 CALL OMIT(IGDS6)
      CALL INCL(IGDSC)
      CALL INCL(IGDS5)
      CALL INCL(IGDS5,NULLV,IKEY3)
      CALL INCL(IGDS5,NULLV,IKEY1)
      GO TO 10
C
C     BRING UP PLOT SELECTED IN PLOT DICTIONARY
C
  300 CONTINUE
      IF(INTAR(4)-ND3)310,210,310
  310 CONTINUE
      CALL BUFIN(I2250,INTAR(4),IRC)
      CALL INCL(IGDSC)
      IF(IRC)320,330,320
  320 WRITE(NOUT,1010)IRC
      GO TO 10
```

```
  330 CALL INCL(IGDS5,NULLV,IKEY2)
      GO TO 10
C
C     RETURN FROM A RECALLED DISPLAY
C
  400 CONTINUE
      CALL BUFIN(I2250,ND3,IRC)
      IF(IRC)320,100,320
C
C     BRING UP NEXT PAGE OF DICTIONARY
C
  250 CONTINUE
      CALL OMIT(IGDS6)
      DO 255 I=ND1,ND2
  255 CALL OMIT(IGDS6,I,NULLV)
      IF(ND2 .EQ. ND3)GO TO 103
      ND1=ND2+1
      IF((ND3-ND2)-20)265,270,270
  265 ND2=ND3
      GO TO 104
  270 ND2=ND2+20
      GO TO 104
 1000 FORMAT(' ATTEN:  PROGRAM TERMINATED BY ''TERMINATE JOB'' OPTION')
 1010 FORMAT(' ***BUFFER WRITE FAILED - R.C. = ',Z8)
 1020 FORMAT(' *GMODEL CHANGE ',Z2)
      END
```

```
H014     TITLE '    '
H014CG   START
*        WRITTEN BY C G HOOKS          IBM HUNTSVILLE
*
*                          CALL BCDWD (WORD, N1, N2, ... )
*
*                                BCDWD CREATES WORD FROM THE DECIMAL EQUI-
*                                      VALENT OF CHARACTERS IN N1, N2, ETC.
*                                      N1 WILL BE THE FIRST CHARACTER LEFT
*                                      ADJUSTED. FOR FULL WORDS THE NUMBER
*                                      OF CHARACTERS MAY BE 1 TO 4. FOR
*                                      DOUBLE PRECISION 1 TO 8. THE WORD
*                                      WILL ONLY BE ALTERED FOR AS MANY
*                                      CHARACTERS AS APPEAR AS ARGUMENTS
*
         ENTRY BCDWD
         USING *,15
BCDWD    BC    15,GO
         DC    X'06'
         DC    CL5'BCDWD'
GO       STM   14,12,12(13)
         L     11,0(0,1)               R11 = ADDRESS OF WORD
LOOP     L     10,4(0,1)               R10 = ADDRESS OF CHARACTER
         MVC   0(1,11),3(10)
         LTR   10,10                   SET CONDITION CODE
         BC    4,FINISH                BRANCH ON LAST ARGUMENT TO FINISH
         A     11,ONE
         A     1,FOUR
         BC    15,LOOP                 BRANCH FOR ANOTHER CHARACTER
FINISH   LM    14,12,12(13)
         SR    15,15
         MVI   12(13),X'FF'
         BCR   15,14
ONE      DC    F'1'
FOUR     DC    F'4'
         END
```

```
      SUBROUTINE CHANGE(Y,DPY)
      REAL * 4 X(92),    Y(366)
C        KWM STORES DATA IN THE ORDER JAN.(YEAR 2) TO SEPT (YEAR 2)
C        FOLLOWED BY OCT.(YEAR 1) TO DEC.(YEAR 1).  FEB. 29 IS PLACED
C        IN DAY 366.
C        CHANGE REARRANGES THE DATA ARRAY TO STORE THE DATA IN NORMAL
C        SEQUENCE.  OCT.(YEAR 1) TO  SEPT.(YEAR 2).
      J = 1
      DO 10 I=274,365
      X(J) = Y(I)
   10 J = J + 1
      K = 365
      J = 273
      IF(DPY .NE. 366) GO TO 15
C        IF LEAP YEAR MOVE FEB. 29TH
      TEMP = Y(366)
      K = 366
   15 CONTINUE
      DO 20 I=1,214
      Y(K) = Y(J)
      K = K - 1
   20 J = J - 1
      IF(DPY .EQ. 366) Y(152) = TEMP
      DO 25 I=1,59
      J = I + 92
   25 Y(J) = Y(I)
      DO 30 I=1,92
   30 Y(I) = X(I)
      RETURN
      END
```

```
      SUBROUTINE CONVER(X,A,IR,NUM)
      REAL CON(24)
      DATA CON/100000.,8.,10000.,7.,1000.,6.,100.,5.,10.,4.,1.,3.,.1,
     12.,.01,2.,.001,2.,.0001,2.,.00001,2.,.000001,2./
      INTEGER*4 A(2),B(2),IR,NUM
      DATA B/2*4H    /
      EQUIVALENCE (Z,IX)
      Z=X
      F=Z
      IF(IR.EQ.1) F=IX
      IS=64
      IF (F.LT.0.0) IS=96
      IF (IS.EQ.96)  F=-F
      M=2
      IF ((IR.EQ.1).AND.(NUM.LE.4)) M=1
      DO 9 I=1,M
      A(I)=B(I)
    9 CONTINUE
      L=1
      IF ((F-100000.).GE.0.) GO TO 15
      L=3
      IF ((F-10000.).GE.0.) GO TO 15
      L=5
      IF ((F-1000.).GE.0.) GO TO 15
      L=7
      IF ((F-100.).GE.0.) GO TO 15
      L=9
      IF ((F-10.).GE.0.) GO TO 15
      L=11
      IF ((F-1.).GE.0.)  GO TO 15
      IF (((F-0.) .EQ.0.) .AND. (IR .EQ.1)) GO TO 23
      L=13
   15 ICNT=CON(L+1)
      CALL ISCHAR(IS,1,A)
      NUMA=NUM+2
      DO 10 I=2,NUMA
      J2=0
      J1=F/CON(L)
      F=F-J1*CON(L)
      J1=J1+240
      IF (((ICNT-I).EQ.0).AND.(IR.EQ.1)) GO TO 12
      IF ((ICNT-I).EQ.0) J2=75
      IF (J2.EQ.75) CALL ISCHAR(J2,I,A)
      IF (J2.EQ.75)  I=I+1
      CALL ISCHAR(J1,I,A)
      L=L+2
   10 CONTINUE
      GO TO 12
   23 J1=240
      CALL ISCHAR(J1,1,A)
   12 RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C     SUBROUTINE CORREL IS USED TO CALCULATE CORRELATION COEFFICIENT
C        X1 =  ARRAY 1   ,   X2 = ARRAY2  , IV = NO.OF VARIABLES IN ARRAY
C        ANS =  ANSWER OF CORRELATION COEFFICIENT
C
      SUBROUTINE CORREL (X1, X2, IV, ANS)
C
      DIMENSION  X1(1), X2(1)
      REAL*4  X1, X2, TX1, TX2, MX1, MX2, DX1, DX2, AX1, AX2, AX3, ANS
C
C
       AX1 = 0.0
       AX2 = 0.0
       AX3 = 0.0
       DX1 = 0.0
       DX2 = 0.0
       MX1 = 0.0
       MX2 = 0.0
       TX1 = 0.0
       TX2 = 0.0
C
C     DETERMINE THE MEAN OF ARRAY X1, X2
C
       DO 100 I = 1,IV
       TX1 = TX1 + X1(I)
  100 TX2 = TX2 + X2(I)
      MX1 = TX1 / IV
      MX2 = TX2 / IV
C
      DO 101 I = 1,IV
      DX1 = X1(I) - MX1
      DX2 = X2(I) - MX2
      AX1 = AX1 + DX1*DX1
      AX2 = AX2 + DX2*DX2
  101 AX3 = AX3 + DX1*DX2
      ANS = AX3 / SQRT(AX1 * AX2)
      RETURN
      END
```

```
      SUBROUTINE DAYNXT(DAY,DPY)
C   DETERMINES NUMBER OF NEXT DAY OF THE YEAR
      INTEGER   DAY,DPY
      DAY = DAY + 1
      IF(DAY .EQ. 366) DAY = 1
      IF(DAY .EQ. 60 .AND. DPY .EQ. 366) DAY = 366
      IF(DAY .EQ. 367) DAY = 60
      RETURN
      END
```

```
      SUBROUTINE DAYOUT(VDCY,MEDWY,DPY)
C   PRINTS TABLE OF DAILY VALUES
      DIMENSION  MEDWY(12),VDCY(366),VDMD(12)
      INTEGER  DATE,DAY,DPY
  100 WRITE(6,1)
    1 FORMAT (7X,'DAY',7X,'OCT',5X,'NOV',5X,'DEC',5X,'JAN',5X,'FEB',
     15X,'MAR',5X,'APR',5X,'MAY',5X,'JUNE',4X,'JULY',4X,'AUG',5X,
     2'SEPT',8X,'ANNUAL')
      MEDWY(3) = 0
      DO 104 DATE = 1,28,1
      IF(MOD(DATE,5) .NE. 1) GO TO 102
      DO 101 KMO = 1,12
      DAY = MEDWY(KMO) + DATE
  101 VDMD(KMO) = VDCY(DAY)
      WRITE(6,2) DATE,VDMD(12),(VDMD(KWD), KWD=1,11)
    2 FORMAT(1H0,3X,I6,3X,12F8.1)
      GO TO 104
  102 DO 103 KMO = 1,12
      DAY = MEDWY(KMO) + DATE
  103 VDMD(KMO) = VDCY(DAY)
      WRITE(6,3) DATE,VDMD(12),(VDMD(KWD), KWD = 1,11)
    3 FORMAT(1X,3X,I6,3X,12F8.1)
  104 CONTINUE
      IF(DPY .NE. 366) GO TO 106
      DATE = 29
      TEMP = VDCY(60)
      VDCY(60) = VDCY(366)
      DO 105 KMO = 1,12
      DAY = MEDWY(KMO) + DATE
  105 VDMD(KMO) = VDCY(DAY)
      WRITE(6,3) DATE,VDMD(12),(VDMD(KWD), KWD=1,11)
      GO TO 107
  106 CONTINUE
      WRITE(6,4) VDCY(302),VDCY(332),VDCY(363),VDCY(29),VDCY(88),
     1VDCY(119),VDCY(149),VDCY(180),VDCY(210),VDCY(241),VDCY(272)
    4 FORMAT(1X,7X,2H29,3X,4F8.1,8X,7F8.1)
  107 CONTINUE
  108 WRITE(6,5) VDCY(303),VDCY(334),VDCY(364),VDCY(30),VDCY(89),
     1VDCY(120),VDCY(150),VDCY(181),VDCY(211),VDCY(242),VDCY(273)
    5 FORMAT(1X,7X,2H30,3X,4F8.1,8X,7F8.1)
      WRITE(6,6) VDCY(304),VDCY(365),VDCY(31),VDCY(90),VDCY(151),
     1VDCY(212),VDCY(243)
    6 FORMAT(1H/,7X,2H31,3X,F8.1,8X,2F8.1,8X,F8.1,8X,F8.1,8X,2F8.1)
      MEDWY(3) = 365
      IF(DPY .EQ. 366) VDCY(60) = TEMP
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE DICTRY(TEXT)
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I225)  ,IATN   ,DAREA  ,
     1      NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2      IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3      VSPB   ,VSPL   ,NUUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4      XLL    ,YLL    ,XUR    ,YUF    ,XTIC   ,YTIC   ,XDVAL  ,
     5      YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6      AY     ,BY     ,YLPOS  ,
     7      LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8      FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      LOGICAL*1 GSPIN, DUM(2)
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      INTEGER*4 UNITN,DAREA,NULLV(1),GSP1
      LOGICAL*1 FA,FB,FC,GMODE1,GSPIN
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LOGICAL*1 LINE(49)/47*' ','@',' '/
      LOGICAL*1 TEXT(72)
      REAL*8 WORK(36)
      EQUIVALENCE(RWORK1,IABSDP)
C
C   CREATE DICTIONARY ENTRY
C
      IABSDP=IABS(IDSP)+1
      IF(IDSP)202,205,205
  202 LINE(1)=TEXT(1)
      LINE(2)=TEXT(2)
      LINE(3)=TEXT(3)
      DO 203 I=4,47
  203 LINE(I)=LINE(49)
      GO TO 215
  205 DO 210 I=1,47
  210 LINE(I)=TEXT(I)
  215 CALL PTEXT(IGDS6,LINE,48,IABSDP,NULLV,2,0.,YLPOS)
      CALL EXEC(IGDS6)
      YLPOS=YLPOS-1.5*VSPL
      IF(YLPOS .LT. 376.)YLPOS=RU-2.5*VSPL
      RETURN
      END
```

```
SUBROUTINE DLPO(IPAR,MLN,NOM,OPAR,NCODE)                             DLP00001
INTEGER *4 TYPE/17,LVAL(1)/1/,HVAL(1)/1/,VDL/17,TDATA/1/             DLP00002
INTEGER *4 IPAR(192)                                                 DLP00003
INTEGER *4 OPAR(32)                                                  DLP00004
CALL DVAR(TYPE,IPAR,MLN,NOM,OPAR,LVAL,HVAL,VDL,TDATA,NCODE)          DLP00005
RETURN                                                               DLP00006
END                                                                  DLP00007
```

```
SUBROUTINE DMSG(IPAR,MLN,NOM,NCODE)                                DMSG0001
INTEGER *4 TYPE/4/,LVAL(1)/1/,HVAL(1)/1/,VDL/1/,TDATA/1/           DMSG0002
INTEGER *4 IPAR(192)                                               DMSG0003
INTEGER *4 OPAR(32)                                                DMSG0004
CALL OVAR(TYPE,IPAR,MLN,NOM,OPAR,LVAL,HVAL,VDL,TDATA,NCODE)        DMSG0005
RETURN                                                             DMSG0006
END                                                                DMSG0007
```

```
         TITLE ' UPDATE MERGE ROUTINE '
         EJECT
DSMRG    START 0
OLD      EQU   2
NEW      EQU   3
LTH      EQU   4
R1       EQU   1
         SAVE  (14,12)
         LR    12,15
         USING DSMRG,12
*
**     SET UP TO UPDATE OLD DATA WITH NEW
*
         LM    OLD,LTH,0(R1)        GET I/P PARMS
         L     LTH,0(LTH)
         SPACE 3
*
**     UPDATE 1ST OR NEXT OLD CHARACTER WITH NEW
*
NEXTCK   EQU   *
         CLI   0(NEW),X'40'
         BE    UPDPTR
         CLI   0(NEW),C'$'
         BE    MOVEBLK
         MVC   0(1,OLD),0(NEW)
         B     UPDPTR
MOVEBLK  EQU   *
         MVI   0(OLD),X'40'
UPDPTR   EQU   *
         LA    OLD,1(OLD)
         LA    NEW,1(NEW)
         BCT   LTH,NEXTCK           UPDATE POINTER
         SPACE 3
*
**     RETURN TO CALLER
*
         RETURN (14,12),RC=0
         END
```

```
         TITLE ' BDAM DATA SET READ ROUTINE '
         EJECT
DCB      EQU   2
AREA     EQU   3
BLOCK    EQU   4
DSREAD   SETUP 12,PARM=11
         LM    DCB,BLOCK,0(R11)     GET I/P PARMS
         EJECT
*
**     READ DATA BLOCK FROM APPLICABLE DATA SET
*
         READ  RDECB,DI,(DCB),(AREA),'S',0,(BLOCK) READ BLOCK OF DATA
         SPACE 3
*
**     WAIT ON READ TO COMPLETE
*
         WAIT  ECB=RDECB            WAIT ON READ TO COMPLETE
         TM    RDECB+1,255          WAS THERE AN ERROR?
         BZ    DSRET                NO, GO TO RETURN
         SPACE 3
*
**     ABEND WITH A DUMP
*
         ABEND 37,DUMP
         SPACE 3
*
**     RETURN TO CALLER
*
DSRET    FREEUP (14,12),T,RC=0
         END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
         TITLE 'HYDROLOGY DATA BASE UPDATE MODULE'
BLK      EQU   2
COUNT    EQU   5
AREA     EQU   6
BLANKS   EQU   7
DSUPDI   SETUP 12,PARM=11
         EJECT
*
**       OPEN DATA SET TO BE UPDATED
*
         OPEN  (BASEDCB,(UPDAT))
         EJECT
*
**       SET UP TO READ PAGE OF DATA
*
READ     EQU   *
         L     R2,ZERO
         ST    R2,NCODE
         LA    AREA,IPBUF
         L     BLANKS,BLANK
         L     COUNT,F500
BLKAREA  EQU   *
         ST    BLANKS,0(AREA)
         LA    AREA,4(AREA)
         BCT   COUNT,BLKAREA
         SPACE 3
*
**       READ PAGE OF DATA
*
         CALL  DSREAD,(BASEDCB,IPBUF,BLOCK)
         EJECT
*
**       DISPLAY PAGE OF DATA BASE FOR POSSIBLE UPDATING
*
         CALL  DVAR,(TYPE,IPAR,MLN,NOM,OPAR,LVAL,HVAL,VDL,TDATA,NCODE)
         EJECT
*
**       WRITE UPDATED PAGE TO DATA BASE
*
         CALL  DSWRT,(BASEDCB,IPBUF,BLOCK)
         SPACE 3
         L     R2,NCODE         DETERMINE STATUS
         S     R2,F100
         BZ    NEXT                  CONTINUE PAGING
         S     R2,ONE
         BZ    TERM                  TERMINATE UPDATE
         S     R2,ONE
         BZ    LAST                  BACK PAGE
*
**       RETURN TO CALLER
*
RET      EQU   *
         FREEUP (14,12),T,RC=0
*
NEXT     L     R2,BLOCK-1
         A     R2,ONE
         ST    R2,BLOCK-1
         B     READ             READ NEXT BLOCK
TERM     L     R2,F100
         ST    R2,NCODE
         CALL  DVAR,(TYPE,IPAR,MLN,NOM,OPAR,LVAL,HVAL,VDL,TDATA,NCODE)
         CLOSE (BASEDCB)
         B     RET              TERMINATE
LAST     L     R2,BLOCK-1
         S     R2,ONE
         BNM   LAST1
         L     R2,ZERO
LAST1    ST    R2,BLOCK-1
         B     READ             READ LAST BLOCK
         EJECT
*
**       CONSTANTS AND WORK AREA
*
TYPE     DC    F'3'
MLN      DC    F'80'
NOM      DC    F'20'
LVAL     DC    F'0'
HVAL     DC    F'0'
VDL      DC    F'72'
TDATA    DC    F'1'
NCODE    DC    F'0'
         DC    F'0'
BLOCK    EQU   *-3
BLANK    DC    C'    '
BASEDCB  DCB   DSORG=DA,MACRF=(RI,WI),OPTCD=R,DDNAME=HYDBASE,RECFM=FB
F500     DC    F'500'
IPBUF    DS    500F
IPAR     EQU   IPBUF
OPAR     EQU   IPBUF
ONE      DC    F'1'
ZERO     DC    F'0'
F100     DC    F'100'
         END
```

```
         TITLE ' BDAM DATA SET WRITE ROUTINE      '
         EJECT
DCB      EQU   2
AREA     EQU   3
BLOCK    EQU   4
DSWRT    SETUP 12,PARM=11
         LM    DCB,BLOCK,0(R11)       GET I/P PARMS
         EJECT
*
**     WRITE DATA BLOCK TO APPLICABLE DATA SET
*
         WRITE WRECB,DI,(DCB),(AREA),'S',0,(BLOCK) WRITE BLOCK OF DATA
         SPACE 3
*
**     WAIT ON WRITE TO COMPLETE
*
         WAIT  ECB=WRECB              WAIT ON WRITE TO COMPLETE
         TM    WRECB+1,255            WAS THERE AN ERROR?
         BZ    DSWRET                 NO, GO TO RETURN
         SPACE 3
*
**     ABEND WITH A DUMP
*
         ABEND 38,DUMP
         SPACE 3
*
**     RETURN TO CALLER
*
DSWRET   FREEUP (14,12),T,RC=0
         END
```

```
C**********************************************************************
C*                                                                    *
C*    TITLE:   DISPLAY / OPTION PROCESSING MODULE (DVAR)              *
C*                                                                    *
C*    FUNCTION: DVAR ALLOWS THE CALLER TO COMMUNICATE WITH THE        *
C*              USER(S) VIA THE 2250 DISPLAY CONSOLE TERMINAL.        *
C*             DVAR PROCESSES LIGHT PEN OPTIONS, VARIABLE DATA        *
C*              OPTION(S), AND TUTORIAL TYPE MESSAGE(S), AND          *
C*              DATA SET UPDATING.                                    *
C*              THE USER(S) RESPONSE(S), LIGHT PEN(S) OR KEYBOARD     *
C*              ACTION(S), AS APPLICABLE, ARE RETURNED TO THE         *
C*              CALLING PROGRAM.                                      *
C*                                                                    *
C*    ENTRY POINT:  DVAR   CALLED BY USING PROGRAM                    *
C*    EXIT: RETURN TO CALLING PROGRAM                                 *
C*                                                                    *
C*    PARM LIST AND DEFINITIONS:                                      *
C*       1. TYPE   -  TYPE OF REQUEST - 1 = LP ACTION, 2 = VAR. DATA  *
C*                                      3 = DATA SET UPDATING         *
C*                                      4 = MESSAGES                  *
C*       2. IPAR   -  LIST OF OPTION(S), MESSAGES, OR DATA SET        *
C*                    RECORDS TO BE TO BE DISPLAYED AND/OR UPDATED    *
C*       3. MLN   -  SIZE IN BYTES OF EACH MEMBER IN IPAR ARRAY       *
C*       4. NUM   -  NO. OF IPAR MEMBERS                              *
C*       5. OPAR  -  AN ARRAY USED TO RETURN USER(S) SELECTIONS       *
C*       6. LVAL  -  LO VALUE ARRAY (VARIABLE DATA)                   *
C*       7. HVAL  -  HI VALUE ARRAY (VARIABLE DATA)                   *
C*       8. VDL   -  MAX. CHARS. USER CAN ENTER FROM 2250 KEYBOARD    *
C*       9. TDATA -  LEGAL VAR. DATA - FLT. PT. = 1,   INTG. = 2      *
C*      10. NCODE -  RETURN CODE                                      *
C*                                                                    *
C**********************************************************************
      SUBROUTINE DVAR(TYPE,IPAR,MLN,NUM,OPAR,LVAL,HVAL,VDL,TDATA,NCODE)
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,DAREA  ,
     1       NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2       IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3       VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4       XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5       YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6       AY     ,BY     ,YLPOS  ,
     7       LOGXSW,LOGYSW,NPTF   ,NPGF   ,ERRHD  ,FA     ,FB     ,
     8   FC     ,GMODE1,GSPIN  ,DUM    ,IPFCAL,IFLAG
      LOGICAL*1 GSPIN, DUM(2)
      INTEGER*4 GSP1,UNITN,DAREA,NULLV(1),STAR(19)
      REAL*8 IDF(9),WORK(36)
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6),GLAB(16)
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LOGICAL*1 FA,FB,FC,GMODE1
      EQUIVALENCE (WORK(1),STAR(1)),(WORK(11),GLAB(1))
      EQUIVALENCE  (NULL,NULLV)
      COMPLEX *16 HDR(21)/'SELECT OPTION(S)',' OR CONTINUE      ',
     C                   '                ','                  ',
     C                   'SELECT VARIABLE ','TO UPDATE         ',
     C                   '                ','                  ',
     C                 'SELECT LINE TO B','E UPDATED         ',
     C                   '                ','                  ',
     C                   'SELECT CONTINUE ','                  ',
     C                   'TO RESUME       ','                  ',
     C                   '                ','                  ',
     C                   '****  INVALID - ','ENTER AGAIN   ****',
     C                   '*               ','                  ',
     C                   'ENTER FLT. PT. O','R INTEGER AS APP',
     C                   'LICABLE         ','                  ',
     C                   '                ','                  ',
     C                   '                ','/'
      COMPLEX *16 BLANKS(5)/'                ','                  ',
     C                   '                ','/'
      COMPLEX *16  BLINE(5)/'||||||||||||||||','||||||||||||||||||',
     C                   '||||||||||||||||','/'
      COMPLEX *16  RBUF(5)/'                ','                  ',
     C                   '                ','/'
      INTEGER*4 PREFIX(32)/' 1. ',' 2. ',' 3. ',' 4. ',
     C                     ' 5. ',' 6. ',' 7. ',' 8. ',
     C                     ' 9. ','10. ','11. ','12. ',
     C                     '13. ','14. ','15. ','16. ',
     C                     '17. ','18. ','19. ','20. ','21. ','22. ',
     C                     '23. ','24. ','25. ','26. ',
     C                     '27. ','28. ','29. ','30. ','31. ','32. '/
      COMPLEX *16 SUFFIX/'----------------'/
      COMPLEX *16   FOOT(2)/'        CONTINUE','----@             '/
      COMPLEX *16 FOOT1(2)/'       TERMINAT','E----@             '/
      COMPLEX *16 FOOT2(2)/'       BACKWARD','----@             '/
      INTEGER *2 DISP/1/
      INTEGER *4 ECODE
      INTEGER*4 IPAR(1)
      INTEGER *2 LP/'@ '/
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      INTEGER *4 LPCA(10)
      INTEGER *4 MSG(7)
      INTEGER *4 TYDATA
      INTEGER *4 TYPE
      INTEGER *4 TDATA
      INTEGER *4 VDL
      INTEGER *4 Y
      INTEGER *4 Z3,Z4
      REAL *4 HVAL(32)
      REAL *4 LVAL(32)
      REAL*4     OPAR(1)
      EQUIVALENCE(IALEV,IATN)
      EQUIVALENCE (MIDSP,IDSP)
      REAL *8 VDATA/'||||||||'/
      REAL *4 Z1,Z2
      EQUIVALENCE (X,Y)
      EQUIVALENCE    (Z1,Z3),(Z2,Z4)
C*
C**  USER'S OPTION(S) PROCESSING
C*
10000 CONTINUE
 1    FORMAT('0 DVAR DISLAY ACTION - TYPE =',I4,' TDATA =',I4)
      IF (IFLAG.EQ.1) GO TO 11001
C   INITIALIZE GRAPHIC SERVICES
      CALL INGSP(GSP1,NULLV)
C   SPECIFY SUBROUTINE LINK/LOAD STATUS
      CALL SPEC(GSP1,2,17,19,26,-28,53,-56)
      CALL SPEC(GSP1,1,7,8,16,29,-31,37,-40)
C   INITIALIZE THE 2250
      CALL INDEV(GSP1,UNITN,I2250)
C   INITIALIZE GRAPHIC DATA SETS
      CALL INGDS(I2250,IGDSC,3072)
      CALL INGDS(I2250,IGDS5,128)
      CALL INGDS(I2250,IGDS6,512)
C   SET NO SCISSIORING FOR ALL DATA SETS
      CALL SSCIS(IGDSC,3)
      CALL SSCIS(IGDS5,3)
      CALL SSCIS(IGDS6,3)
C*
C**  SET UP TO BUILD DATA SET(S)
C*
11001 CONTINUE
      IF (TYPE .EQ. 4) GO TO 2000
      CALL INGDS(I2250,IGDS,1536)
      CALL INGDS(I2250,MSG(1),128,NULL,MSG(2),MSG(3),MSG(4),MSG(5),
     C                    MSG(6),MSG(7))
      CALL SGRAM(IGDS,2)
      CALL SDATM(IGDS,3)
      CALL CRATL(I2250,IALEV)
      CALL SLPAT(IGDS,1)
      NCHMOD = 4
      NX = 48
      NY = 20
      IF (TYPE.NE.3) GO TO 11010
      NCHMOD = 3
      NX = 72
      NY = 49
11010 CONTINUE
      CALL SCHAM(IGDS,NCHMOD)
      CALL SDATL(IGDS,0,0,NX,NY)
C*
C**  BUILD MSG DATA SET(S )
C*
      DO  11002 N=1,7
      CALL SCHAM(MSG(N),NCHMOD-2)
      CALL SGRAM(MSG(N),2)
      CALL SDATM(MSG(N),3)
      CALL SGDSL(MSG(N),0,NY-1,NX,NY,0,0,NX,NY)
      CALL PTEXT(MSG(N),HDR(3*N-2),48,1,NULL,1,0,NY)
11002 CONTINUE
C*
C**  SET UP TO COMMUNICATE WITH 2250 OPERATOR
C*
11003 CONTINUE
      DO 11004 N=1,6
      CALL INCL(MSG(N))
11004 CONTINUE
      CALL PTEXT(IGDS,FOOT ,32,100,NULL,1,0,6)
      IF (TYPE.NE.3) GO TO 5
      CALL PTEXT(IGDS,FOOT1,32,101,NULL,1,0,4)
      CALL PTEXT(IGDS,FOOT2,32,102,NULL,1,0,2)
 5    CONTINUE
      CALL PTEXT(IGDS,BLANKS,NX,50,NULL,1,0,1)
      ECODE = 4
      GO TO (10,10,3000,2000), TYPE
      GO TO 1040
 10   CONTINUE
      ECODE = 8
      IF (MLN.LE.0.OR.MLN.GT.12) GO TO 1040
```

```
      ECODE = 12
      IF (NOM.LE.0.OR.NOM.GT.32) GO TO 1040
      IF (TYPE.EQ.1) GO TO 20
      ECODE = 16
      IF (VDL.LE.0.OR.VDL.GT.6) GO TO 1040
      ECODE = 20
      IF (TDATA.LE.0.OR.TDATA.GT.2) GO TO 1040
C*
C**   SET UP TO ACCESS I/P ARRAY MEMBERS
C*
20    CONTINUE
      ECODE = 0
      X = NOM
      X = (X/2.0) + 0.75
      M = X
50    CONTINUE
      X = MLN
      X = (X/4.0) + 0.75
      I = X
      LTH = 4 * I
C*
C**   BUILD VARIABLE OR OPTION(S) DATA SET
C*
      NCV = 0
      DO 100 L=1,M,1
      NCV = NCV + 1
      CALL PTEXT(IGDS,PREFIX(L),3,NULL,NULL,1,0,19-NCV)
      CALL PTEXT(IGDS,IPAR(I*(L-1)+1),LTH,NULL,NULL,1,4,19-NCV)
      IF (MLN.GE.11) GO TO 99
      CALL PTEXT(IGDS,SUFFIX,(11-MLN),NULL,NULL,1,MLN+4,19-NCV)
99    CONTINUE
      CALL PTEXT(IGDS,LP,2,L,NULL,1,16,19-NCV)
100   CONTINUE
      IF (M.EQ.1.AND.NOM.EQ.1) GO TO 299
      ICV = NCV
      M = M+1
      DO 200 L=M,NCM,1
      NCV = NCV + 1
      CALL PTEXT(IGDS,PREFIX(L),3,NULL,NULL,1,25,ICV+19-NCV)
      CALL PTEXT(IGDS,IPAR(I*(L-1)+1),LTH,NULL,NULL,1,29,ICV+19-NCV)
      IF (MLN.GE.11) GO TO 199
      CALL PTEXT(IGDS,SUFFIX,(11-MLN),NULL,NULL,1,MLN+29,ICV+19-NCV)
199   CONTINUE
      CALL PTEXT(IGDS,LP,2,L,NULL,1,41,ICV+19-NCV)
200   CONTINUE
C*
C**   DISPLAY MAIN OPTIONS
C*
299   CONTINUE
      CALL SALRM(I2250)
300   CONTINUE
      CALL EXEC(IGDS)
      CALL EXEC(MSG(TYPE))
      CALL ENATN(IALEV,34)
C*
C**   PROCESS LP ACTION
C*
      CALL RQATN(IALEV,NCODE,2,LPCA,34)
      CALL DSATN(IALEV,34)
      LPSEL = LPCA(4)
      IF (LPSEL) 300,300,310
310   CONTINUE
      IF (LPSEL.EQ.100 .OR. LPSEL.EQ.101 .OR. LPSEL.EQ.102) GO TO 1050
      IF (TYPE.EQ.4.OR.LPSEL.GT.NOM) GO TO 300
320   CONTINUE
      WRITE (6,330)  LPSEL
330   FORMAT ('- LP SEL. = ',I2)
C*
C**   DISPLAY VARIABLE I/P AREA
C*
400   CONTINUE
      GO TO (620,405,402), TYPE
402   CONTINUE
      LPS = 50
      CALL PTEXT(IGDS,BLINE,VDL,LPS,NULL,3,0,NY-(2*LPSEL)-1)
      GO TO 411
405   CONTINUE
      IF (TYPE.NE.2) GO TO 620
      CALL EXEC(MSG(6))
      LPS = 50
      N = 18
      K = 19 - LPSEL
      IF (LPSEL.EQ.1.OR.LPSEL.LT.M) GO TO 410
      N = 43
      K = M + 18 - LPSEL
410   CONTINUE
      CALL PTEXT(IGDS,VDATA,VDL,LPS,NULL,3,N,K)
411   CONTINUE
      CALL INCL(IGDS,LPS)
```

```
      CALL ICURS(IGDS,LPS,NULL,1)
      CALL EXEC(IGDS)
      CALL ENATN(IALEV,32,34)
      CALL RQATN(IALEV,MCODE,2,LPCA,32,34)
      CALL DSATN(IALEV,32,34)
      CALL OMIT(IGDS,LPS)
      IF (MCODE.EQ.32) GO TO 500
      IF (LPCA(4)) 411,411,420
420   CONTINUE
      IF (LPCA(4).EQ.100) GO TO 1050
      IF (LPCA(4).GT.NOM) GO TO 411
      LPSEL = LPCA(4)
      GO TO 320
C*
C**  GET VARIABLE I/P DATA
C*
500   CONTINUE
      DO 505 N=1,5
      RBUF(N) = BLANKS(1)
505   CONTINUE
      NC = VDL
      NS = -1 * NC
      CALL GSPRD(IGDS,RBUF,NS,1,NTCODE,LPS,NULL)
      WRITE (6,510)  RBUF
510   FORMAT (' VAR. ENTRY = ',10A8)
      IF (NTCODE) 512,513,514
512   NTCODE = -1 * NTCODE
514   NC = NTCODE
C*
C**  CONVERT DATA TO REAL OR INTEGER AS APPLICABLE
C*
513   CONTINUE
      IF (TYPE.EQ.3) GO TO 700
      CALL CVRTIN(RBUF,NC,X,IERR,TYDATA)
      IF (IERR.NE.0.OR.TDATA.NE.TYDATA) GO TO 1030
      IF (TDATA.EQ.2) GO TO 610
600   CONTINUE
      IF (X.LT.LVAL(LPSEL).OR.X.GT.HVAL(LPSEL))  GO TO 1030
      GO TO 630
610   CONTINUE
      Z1 = LVAL(LPSEL)
      Z2 = HVAL(LPSEL)
      IF (Y.LT.Z3.OR.Y.GT.Z4)  GO TO 1030
      GO TO 630
620   CONTINUE
      Y = 1
      CALL OMIT(IGDS,LPSEL)
630   CONTINUE
      OPAR(LPSEL) = X
      GO TO 300
C*
C**   UPDATE LINE OF DATA
C*
700   CONTINUE
      CALL DSMRG(IPAR(I*(LPSEL-1)+1),RBUF,NC)
      CALL PTEXT(IGDS,IPAR(I*(LPSEL-1)+1),MLN,LPSEL,NULL,3,0,NY-2*LPSEL)
      GO TO 300
C*
C**  BUILD TEXT DATA SET
C*
2000  CONTINUE
      ECODE = 8
      IF (MLN .LE. 0 .OR. MLN .GT. 80) GO TO 1040
      ECODE = 16
      IF(NOM.LE.0.OR.NOM.GT.49)  GO TO 1040
      ECODE = 0
      IRECAL=0
      IFLAG = 1
      X = MLN
      X = (X/4.0) + 0.75
      I = X
      CALL SHIFTI
      CALL DICTRY(OPAR(1))
      DO 2003 N = 1, NOM
      X = 3920.0 - N*80
      CALL PTEXT(IGDSC,IPAR(I*(N-1)+1),MLN,NULL,NULL,1,0.0,X)
 2003 CONTINUE
 9999 FORMAT(' DVAR TERMINATE RECALL: TYPE =',I4,' TDATA =',I4,
     1       ' MLN =',I4,' NOM =',I4,' NCODE =',I4)
      GO TO 1052
C*.
C** SET UP TO UPDATE DATA SET
C*
3000  CONTINUE
      IF (NCODE.NE.0) GO TO 1050
      ECODE = 8
      IF (MLN.LE.0.OR.MLN.GT.80) GO TO 1040
      ECODE = 16
      IF (NOM.LE.0.OR.NOM.GT.22)  GO TO 1040
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IF (VDL.LE.0.OR.VDL.GT.72) GO TO 1040                         DVAR0359
      ECODE = 0                                                      DVAR0360
      X = MLN                                                        DVAR0361
      X = (X/4.0) + 0.75                                             DVAR0362
      I = X                                                          DVAR0363
      DO 3010 N=1,NOM,1                                              DVAR0364
      CALL PTEXT(IGDS,IPAR(I*(N-1)+1),MLN,N,NULL,1,0,NY-2*N)         DVAR0365
 3010 CONTINUE                                                       DVAR0366
      GO TO 299                                                      DVAR0367
C*                                                                   DVAR0368
C**  CONVERSION ROUTINE ERROR RETURN                                 DVAR0369
C*                                                                   DVAR0370
 1030 CONTINUE                                                       DVAR0371
      CALL EXEC(MSG(5))                                              DVAR0372
      CALL INCL(IGDS,LPS)                                            DVAR0373
      WRITE (6,1031)                                                 DVAR0374
      GO TO 411                                                      DVAR0375
 1031 FORMAT (' ***  INVALID 2250 ENTRY  *** ')                      DVAR0376
C*                                                                   DVAR0377
C**   RETURN TO CALLER                                               DVAR0378
C*                                                                   DVAR0379
 1040 CONTINUE                                                       DVAR0380
      WRITE(6,1041) NCODE                                            DVAR0381
 1041 FORMAT(' INVALID I/P PARM(S)',I4)                              DVAR0382
 1050 CONTINUE                                                       DVAR0383
      IFLAG = 1                                                      DVAR0384
      IRECAL=0                                                       DVAR0385
      CALL TMGDS(IGDS)                                               DVAR0386
      CALL EXEC(MSG(7))                                              DVAR0387
      DO 1051  N=1,7                                                 DVAR0388
      CALL TMGDS(MSG(N))                                             DVAR0389
 1051 CONTINUE                                                       DVAR0390
 1052 CONTINUE                                                       DVAR0391
      NCODE = ECODE                                                  DVAR0392
      IF (TYPE.EQ.3) NCODE = NCODE + LPSEL                           DVAR0393
 9997 FORMAT(' DVAR TERMINATE: NCODE =',I4)                          DVAR0395
      RETURN                                                         DVAR0396
      END                                                            DVAR0397
```

```
      SUBROUTINE EVPDAY(DPET, EMAET)
C   DETERMINES DATED PAN EVAPORATION TOTALS
      DIMENSION  DPET(366)
      INTEGER DAY
      DO 100  DAY = 1,5
  100 DPET(DAY) = 0.00060*EMAET
      DPET(  6) = 0.00059*EMAET
      DPET(  7) = DPET(  6)
      DO 101 DAY = 8,10
  101 DPET(DAY) = 0.00058*EMAET
      DO 102 DAY = 11,16
  102 DPET(DAY) = 0.00057*EMAET
      DPET( 17) = DPET(  8)
      DO 103 DAY = 18,20
  103 DPET(DAY) = DPET(  6)
      DO 104 DAY = 21,32
  104 DPET(DAY) = DPET(  1)
      DPET( 33) = 0.00061*EMAET
      DO 105 DAY = 34,38
  105 DPET(DAY) = 0.00062*EMAET
      DPET( 39) = 0.00063*EMAET
      DPET( 40) = DPET( 39)
      DPET( 41) = 0.00064*EMAET
      DPET( 42) = 0.00065*EMAET
      DPET( 43) = 0.00066*EMAET
      DO 106 DAY = 44,50
  106 DPET(DAY) = 0.00067*EMAET
      DO 107 DAY = 51,55
  107 DPET(DAY) = 0.00068*EMAET
      DPET( 56) = 0.00069*EMAET
      DO 108 DAY = 57,61
  108 DPET(DAY) = 0.00070*EMAET
      DPET( 62) = 0.00071*EMAET
      DPET( 63) = 0.00072*EMAET
      DPET( 64) = DPET( 63)
      DPET( 65) = 0.00073*EMAET
      DPET( 66) = 0.00074*EMAET
      DPET( 67) = 0.00075*EMAET
      DPET( 68) = 0.00076*EMAET
      DPET( 69) = 0.00077*EMAET
      DPET( 70) = DPET( 69)
      DPET( 71) = 0.00078*EMAET
      DPET( 72) = DPET( 71)
      DPET( 73) = 0.00079*EMAET
      DPET( 74) = DPET( 73)
      DPET( 75) = 0.00080*EMAET
      DPET( 76) = 0.00081*EMAET
      DPET( 77) = 0.00082*EMAET
      DPET( 78) = 0.00084*EMAET
      DPET( 79) = 0.00086*EMAET
      DPET( 80) = 0.00088*EMAET
      DPET( 81) = 0.00090*EMAET
      DPET( 82) = 0.00092*EMAET
      DPET( 83) = 0.00094*EMAET
      DPET( 84) = 0.00097*EMAET
      DPET( 85) = 0.00099*EMAET
      DPET( 86) = 0.00102*EMAET
      DPET( 87) = 0.00106*EMAET
      DPET( 88) = 0.00109*EMAET
      DPET( 89) = 0.00113*EMAET
      DPET( 90) = 0.00118*EMAET
      DPET( 91) = 0.00122*EMAET
      DPET( 92) = 0.00128*EMAET
      DPET( 93) = 0.00132*EMAET
      DPET( 94) = 0.00137*EMAET
      DPET( 95) = 0.00142*EMAET
      DPET( 96) = 0.00147*EMAET
      DPET( 97) = 0.00151*EMAET
      DPET( 98) = 0.00157*EMAET
      DPET( 99) = 0.00163*EMAET
      DPET(100) = 0.00168*EMAET
      DPET(101) = 0.00173*EMAET
      DPET(102) = 0.00178*EMAET
      DPET(103) = 0.00185*EMAET
      DPET(104) = 0.00193*EMAET
      DPET(105) = 0.00201*EMAET
      DPET(106) = 0.00208*EMAET
      DPET(107) = 0.00214*EMAET
      DPET(108) = 0.00221*EMAET
      DPET(109) = 0.00227*EMAET
      DPET(110) = 0.00234*EMAET
      DPET(111) = 0.00241*EMAET
      DPET(112) = 0.00249*EMAET
      DPET(113) = 0.00256*EMAET
      DPET(114) = 0.00262*EMAET
      DPET(115) = 0.00268*EMAET
      DPET(116) = 0.00276*EMAET
      DPET(117) = 0.00281*EMAET
      DPET(118) = 0.00287*EMAET
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
DPET(119) = 0.00293*EMAET
DPET(120) = 0.00299*EMAET
DPET(121) = 0.00305*EMAET
DPET(122) = 0.00310*EMAET
DPET(123) = 0.00317*EMAET
DPET(124) = 0.00322*EMAET
DPET(125) = 0.00328*EMAET
DPET(126) = 0.00333*EMAET
DPET(127) = 0.00338*EMAET
DPET(128) = 0.00344*EMAET
DPET(129) = 0.00348*EMAET
DPET(130) = 0.00354*EMAET
DPET(131) = 0.00359*EMAET
DPET(132) = 0.00365*EMAET
DPET(133) = 0.00370*EMAET
DPET(134) = 0.00374*EMAET
DPET(135) = 0.00378*EMAET
DPET(136) = 0.00382*EMAET
DPET(137) = 0.00387*EMAET
DPET(138) = 0.00391*EMAET
DPET(139) = 0.00394*EMAET
DPET(140) = 0.00399*EMAET
DPET(141) = 0.00402*EMAET
DPET(142) = 0.00407*EMAET
DPET(143) = 0.00411*EMAET
DPET(144) = 0.00417*EMAET
DPET(145) = 0.00420*EMAET
DPET(146) = 0.00426*EMAET
DPET(147) = 0.00430*EMAET
DPET(148) = 0.00436*EMAET
DPET(149) = 0.00440*EMAET
DPET(150) = 0.00445*EMAET
DPET(151) = 0.00450*EMAET
DPET(152) = 0.00455*EMAET
DPET(153) = 0.00460*EMAET
DPET(154) = 0.00466*EMAET
DPET(155) = 0.00470*EMAET
DPET(156) = 0.00473*EMAET
DPET(157) = 0.00478*EMAET
DPET(158) = 0.00482*EMAET
DPET(159) = 0.00487*EMAET
DPET(160) = 0.00491*EMAET
DPET(161) = 0.00495*EMAET
DPET(162) = 0.00500*EMAET
DPET(163) = 0.00504*EMAET
DPET(164) = 0.00508*EMAET
DPET(165) = 0.00510*EMAET
DPET(166) = 0.00512*EMAET
DPET(167) = 0.00514*EMAET
DPET(168) = 0.00515*EMAET
DPET(169) = 0.00517*EMAET
DPET(170) = 0.00519*EMAET
DPET(171) = 0.00520*EMAET
DPET(172) = 0.00521*EMAET
DPET(173) = DPET(172)
DPET(174) = DPET(172)
DPET(175) = 0.00522*EMAET
DPET(176) = 0.00523*EMAET
DPET(177) = 0.00524*EMAET
DPET(178) = 0.00525*EMAET
DPET(179) = 0.00527*EMAET
DPET(180) = 0.00528*EMAET
DPET(181) = DPET(180)
DPET(182) = 0.00529*EMAET
DPET(183) = 0.00530*EMAET
DPET(184) = DPET(183)
DPET(185) = 0.00531*EMAET
DPET(186) = 0.00532*EMAET
DPET(187) = 0.00533*EMAET
DPET(188) = 0.00534*EMAET
DPET(189) = DPET(188)
DPET(190) = 0.00535*EMAET
DPET(191) = 0.00536*EMAET
DPET(192) = 0.00537*EMAET
DPET(193) = 0.00538*EMAET
DPET(194) = DPET(193)
DPET(195) = 0.00539*EMAET
DPET(196) = 0.00540*EMAET
DPET(197) = DPET(196)
DPET(198) = 0.00541*EMAET
DPET(199) = 0.00542*EMAET
DPET(200) = 0.00543*EMAET
DPET(201) = 0.00545*EMAET
DPET(202) = 0.00546*EMAET
DPET(203) = 0.00547*EMAET
DPET(204) = 0.00548*EMAET
DPET(205) = 0.00549*EMAET
DPET(206) = 0.00550*EMAET
DPET(207) = 0.00551*EMAET
```

```
      DPET(208) = 0.00552*EMAET
      DPET(209) = 0.00553*EMAET
      DPET(210) = 0.00555*EMAET
      DPET(211) = 0.00557*EMAET
      DPET(212) = 0.00558*EMAET
      DPET(213) = 0.00560*EMAET
      DPET(214) = DPET(213)
      DPET(215) = C.00561*EMAET
      DPET(216) = 0.00562*EMAET
      DPET(217) = 0.00563*EMAET
      DPET(218) = 0.00565*EMAET
      DPET(219) = 0.00567*EMAET
      DPET(220) = DPET(219)
      DO 109 DAY = 221,226
  109 DPET(DAY) = 0.00568*EMAET
      DO 110 DAY = 227,229
  110 DPET(DAY) = DPET(219)
      DPET(230) = 0.00566*EMAET
      DPET(231) = 0.00564*EMAET
      DPET(232) = DPET(217)
      DPET(233) = DPET(216)
      DPET(234) = DPET(213)
      DPET(235) = 0.00559*EMAET
      DPET(236) = DPET(211)
      DPET(237) = DPET(210)
      DPET(238) = DPET(209)
      DPET(239) = DPET(206)
      DPET(240) = DPET(203)
      DPET(241) = DPET(199)
      DPET(242) = DPET(193)
      DPET(243) = DPET(190)
      DPET(244) = DPET(185)
      DPET(245) = DPET(179)
      DPET(246) = DPET(175)
      DPET(247) = DPET(169)
      DPET(248) = 0.00511*EMAET
      DPET(249) = DPET(163)
      DPET(250) = 0.00497*EMAET
      DPET(251) = 0.00490*EMAET
      DPET(252) = DPET(158)
      DPET(253) = 0.00476*EMAET
      DPET(254) = 0.00468*EMAET
      DPET(255) = 0.00461*EMAET
      DPET(256) = 0.00454*EMAET
      DPET(257) = DPET(150)
      DPET(258) = 0.00437*EMAET
      DPET(259) = 0.00427*EMAET
      DPET(260) = 0.00418*EMAET
      DPET(261) = DPET(142)
      DPET(262) = 0.00397*EMAET
      DPET(263) = DPET(137)
      DPET(264) = 0.00377*EMAET
      DPET(265) = 0.00367*EMAET
      DPET(266) = 0.00356*EMAET
      DPET(267) = 0.00347*EMAET
      DPET(268) = 0.00337*EMAET
      DPET(269) = 0.00329*EMAET
      DPET(270) = DPET(124)
      DPET(271) = 0.00315*EMAET
      DPET(272) = 0.00308*EMAET
      DPET(273) = 0.00303*EMAET
      DPET(274) = 0.00300*EMAET
      DPET(275) = 0.00298*EMAET
      DPET(276) = 0.00294*EMAET
      DPET(277) = 0.00290*EMAET
      DPET(278) = 0.00286*EMAET
      DPET(279) = 0.00283*EMAET
      DPET(280) = 0.00279*EMAET
      DPET(281) = DPET(116)
      DPET(282) = 0.00271*EMAET
      DPET(283) = DPET(115)
      DPET(284) = DPET(114)
      DPET(285) = 0.00259*EMAET
      DPET(286) = 0.00254*EMAET
      DPET(287) = 0.00252*EMAET
      DPET(288) = 0.00247*EMAET
      DPET(289) = 0.00244*EMAET
      DPET(290) = 0.00239*EMAET
      DPET(291) = DPET(110)
      DPET(292) = 0.00230*EMAET
      DPET(293) = 0.00225*EMAET
      DPET(294) = 0.00222*EMAET
      DPET(295) = 0.00217*EMAET
      DPET(296) = 0.00213*EMAET
      DPET(297) = 0.00210*EMAET
      DPET(298) = 0.00206*EMAET
      DPET(299) = 0.00200*EMAET
      DPET(300) = 0.00197*EMAET
```

```
      DPET(301) = 0.00194*EMAET
      DPET(302) = 0.00189*EMAET
      DPET(303) = 0.00186*EMAET
      DPET(304) = 0.00183*EMAET
      DPET(305) = 0.00180*EMAET
      DPET(306) = 0.00177*EMAET
      DPET(307) = 0.00174*EMAET
      DPET(308) = 0.00172*EMAET
      DPET(309) = DPET(100)
      DPET(310) = DPET( 99)
      DPET(311) = 0.00160*EMAET
      DPET(312) = 0.00156*EMAET
      DPET(313) = 0.00152*EMAET
      DPET(314) = 0.00149*EMAET
      DPET(315) = 0.00146*EMAET
      DPET(316) = DPET( 95)
      DPET(317) = 0.00138*EMAET
      DPET(318) = 0.00135*EMAET
      DPET(319) = 0.00131*EMAET
      DPET(320) = 0.00127*EMAET
      DPET(321) = 0.00124*EMAET
      DPET(322) = 0.00120*EMAET
      DPET(323) = DPET( 90)
      DPET(324) = 0.00116*EMAET
      DPET(325) = DPET( 89)
      DPET(326) = 0.00110*EMAET
      DPET(327) = 0.00107*EMAET
      DPET(328) = 0.00104*EMAET
      DPET(329) = DPET( 86)
      DPET(330) = 0.00100*EMAET
      DPET(331) = 0.00098*EMAET
      DPET(332) = 0.00097*EMAET
      DPET(333) = 0.00095*EMAET
      DPET(334) = 0.00093*EMAET
      DPET(335) = DPET( 81)
      DPET(336) = DPET( 80)
      DPET(337) = 0.00087*EMAET
      DPET(338) = DPET( 79)
      DPET(339) = DPET( 78)
      DPET(340) = DPET( 77)
      DPET(341) = DPET( 75)
      DPET(342) = DPET( 73)
      DPET(343) = DPET( 71)
      DPET(344) = DPET( 71)
      DPET(345) = DPET( 69)
      DPET(346) = DPET( 68)
      DPET(347) = DPET( 66)
      DPET(348) = DPET( 63)
      DPET(349) = DPET( 62)
      DPET(350) = DPET( 57)
      DPET(351) = DPET( 57)
      DPET(352) = DPET( 56)
      DO 111 DAY = 353,355
  111 DPET(DAY) = DPET( 51)
      DPET(356) = DPET( 44)
      DPET(357) = DPET( 44)
      DPET(358) = DPET( 42)
      DPET(359) = DPET( 41)
      DPET(360) = DPET( 39)
      DPET(361) = DPET( 34)
      DPET(362) = DPET( 33)
      DO 112 DAY = 363,365
  112 DPET(DAY) = DPET(  1)
      DPET(366) = DPET( 57)
      RETURN
      END
```

```
      BLOCK DATA
      COMMON/GSPD/
     1    WORK        ,GSP1        ,UNITN       ,I2250       ,IATN        ,
     2    OAREA       ,NULLV       ,IGDSC       ,IPTNOW      ,MAXPT       ,
     3    IPT         ,ISYM        ,RWORK1      ,IGDS4       ,IGDS5       ,
     4    IGDS6       ,IDSP        ,RU          ,HSPB        ,HSPL        ,
     5    VSPB        ,VSPL        ,NOUT        ,NXD         ,NYD         ,
     6    XSIZE       ,YSIZF       ,XLL         ,YLL         ,XUR         ,
     7    YUR         ,XTIC        ,YTIC        ,XDVAL       ,YDVAL       ,
     8    IKEY1       ,IKEY2       ,IKEY3       ,IKEY4       ,AX          ,
     9    BX          ,AY          ,BY          ,YLPOS       ,
     A    LOGXSW      ,LOGYSW      ,NPTF        ,NRGF        ,ERRHD       ,
     B      FA       ,FB      ,FC       ,GMODE1,GSPIN ,DUM    ,IRECAL
      LOGICAL*1 DUM(2)
      REAL*8 WORK(36)/10*'********','CONTINUE','--------','---@TABL',
     1  'E OF CON','TENTS--@','RETURN--','---@NEXT',' P&GE--@',
     2     18*'        '/
      REAL*4 RU/4096./    ,HSPB/56./    ,HSPL/84./    ,VSPB/80./    ,
     1       VSPL/120./   ,XLL/896./    ,XUR/3984./   ,YUR/3616./   ,
     2       YLPOS/3796./,RWORK1/1810./
      INTEGER*4 UNITN/10/    ,OAREA/768/   ,NULLV(1)/-5/,IDSP/0/     ,
     1          NOUT/48/     ,NXD/6/       ,NYD/6/       ,GSP1        ,
     2          IPT/36/      ,MAXPT/640/   ,IPTNOW/0/
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LOGICAL*1 FA,FB,FC,GMODE1/.TRUE./,GSPIN/.FALSE./
      END
```

```
C     PROGRAM FINDS THE INTEGRAL OF A GIVEN HYDROGRAPH AND PLOTS THE     INTG0001
C     INPUT DATA.                                                        INTG0002
C                                                                        INTG0003
C     DATA INPUT IN HRS.FT.                                              INTG0004
C     DATA PLOTTED IN HRS.FT. OR HRS.SEC.FT.                             INTG0005
C     DATA INTEGRATED IN DAY.FT. OR DAY.SEC.FT.                          INTG0006
C     IFLAG= 1  TRAPEZOIDAL INTEGRATION SCHEME USED.                     INTG0007
C     IFLAG= 2  SIMPSON'S 1/3 INTEGRATION SCHEME USED.                   INTG0008
C     IPLOT= 0  1403 PRINTER PLOTS ONLY.                                 INTG0009
C     IPLOT= 1   BOTH 1403 PRINTER PLOTS AND SC4020 PLOTS.               INTG0010
C     LOOKUP= -1  NO TABLE LOOKUP REQUIRED FOR CURRENT CASE.             INTG0011
C     LOOKUP=  0  PREVIOUS TABLE REQUIRED FOR CURRENT CASE.              INTG0012
C     LOOKUP=  1  NEW TABLE REQUIRED FOR CURRENT CASE.                   INTG0013
C                                                                        INTG0014
      SUBROUTINE INT(NQ,Q,DT,AREAI,LOOKUP,IPLOT,NFP,MSBDIC,MPDAY,DIFFS, INTG0015
     1DIFFMP,DIFFP,DIFFPP,SUM,DIFFKA,DIFFRP,APREC,DIFFPR,DIFPP,QOUT,     INTG0016
     2QMAX,PHRO)                                                         INTG0017
      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLF,INTG0018
     2KFLAG,IDFLAG,                                                      INTG0019
     1IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,        INTG0020
     2 NCTRI,CTRI,FIRR,RICY,DPSE,EDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,         INTG0021
     3 XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,           INTG0022
     A SATRI,UHFA,                                                       INTG0023
     BMNRD,                                                              INTG0024
     4 FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,             INTG0025
     5 BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,            INTG0026
     6 BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPPC,     INTG0027
     7 IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,THCFS,EPAET,EPER,        INTG0028
     8 TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPOR,OHUS,OHUSIS,OFR,OFRIS,PEIS,   INTG0029
     9 RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,     INTG0030
     A WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,             INTG0031
     B SDEPTH,MULTI,ID,ASP,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,   INTG0032
     XDEPEND(2),VARIN(2),NPTS,JULDT,IYR,TODARY(5,1),                     INTG0033
     YTOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),                 INTG0034
     ZTSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),                   INTG0035
     WTORARY(1,6,1),                                                     INTG0036
     CDRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)             INTG0037
      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,    INTG0038
     1TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,INTG0039
     2EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,INTG0040
     3TMSTFI,T20DFH,T20PRH,TMRTFI,JULDAT,                                INTG0041
     4TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,         INTG0042
     5TRHF,                                                              INTG0043
     5SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS                      INTG0044
      DIMENSION TITLF(18),Q(337),T(337),TABT(18)                         INTG0045
     1          ,FLDT(18),FLDQ(18),IDF(18),TEMP(1540)                    INTG0046
     1          ,YLAB(13)    ,FLD(6),A(10),SQ(500),ANSQ(3),ANSS(3),      INTG0047
     2  NSPTS(6),SUBT(18),DRSF(366),DSSF(366),TITLA(18),CONOPT(16),      INTG0048
     1THSFD(744,3),PEAKS(6),THSFDM(6),PHRS(6),TMRTF(12),TMSTF(15,3),     INTG0049
     A  CTRI(99),FIRR(15),RICY(37),DPSE(366),BTRI(99),                   INTG0050
     A CRFMI(22),DDIW(366),DMNT(366),DMXT(366),DRGPM(366),DRHP(366,24),  INTG0051
     3 EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),    INTG0052
     BDRSGP(366),DPET(366),EDLZS(366),SERA(22),SERR(22),SESF(22),        INTG0053
     CSQER(22),                                                          INTG0054
     4 EMUZC(15,3),EMUZS(15,3),EPCM(12),                                 INTG0055
     6            THSF(24),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),   INTG0056
     7 TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMSE(15,3),           INTG0057
     8             TMSNE(12),TMSTFI(15,3),T20DFH(21),T20PRH(21),         INTG0058
     3TMRTFI(12),JULDAT(6),TFMAXY(366),                                  INTG0059
     B SATRI(99),UHFA(99),APREC(3),QOUT(15)                              INTG0060
      INTEGER  DIFFP,STPTS,PHRS,FLDY(18),STUDY,TITLA,TABT,END            INTG0061
      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY                  INTG0062
      INTEGER FLDP(18),CONOPT,PCHARP(2),PCHARF(3),PCHAR(3),SINDEX,       INTG0063
     1MPDAY(15),FLDQ,FLDT                                                INTG0064
      INTEGER     ANSQ,ANSS,PHRO,SUBT                                    INTG0065
      REAL SUM(3)                                                        INTG0066
      DATA PCHARP/38,50/                                                 INTG0067
      DATA PCHARF/54,57,22/                                              INTG0068
      INTEGER PFCHAR(3)/-230,-233,198/,PPCHAR(2)/-214,226/               INTG0000
      INTEGER PRCHAR(3)                                                  INTG0000
      DATA SUBT/'REF ','R/O ',3*4H    ,' IN ','<TIM',                    INTG0069
     1'E(HR',')> S','IM R','/O  ',3*4H    ,' IN ',3*4H    /              INTG0070
      DATA END/'END '/                                                   INTG0071
      DATA BLK /4H    /                                                  INTG0072
      DATA FLDQ/'H/C ','VS N','/P V','S F/','C HO','UFLY',' R/O',        INTG0073
     *11*4H    /                                                         INTG0074
      DATA FLDP/'HOUR','LY  ','PREC','IPAT','ATIO','N   ','(IN)',        INTG0075
     111*4H    /                                                         INTG0076
      DATA FLDY/'HOUR','LY O','BS A','ND S','IM R','/O (','CFS)',        INTG0077
     111*4H    /                                                         INTG0078
      DATA FLDT/6*4H    ,'TIME','(HRS',')   ',9*4H    /                  INTG0079
      DATA FLD /'E(FT',')   ','    ','F(SE','C.FT','.)  '/               INTG0080
C                                                                        INTG0081
9     IERR=0                                                             INTG0082
      INDEXM=0                                                           INTG0083
      IDIAG=1                                                            INTG0084
      DO   180  I=1,1540                                                 INTG0000
180   TEMP(I)=0.0                                                        INTG0000
      READ(5,500) (IDF(I),I=1,18)                                        INTG0085
```

```
  500 FORMAT(18A4)
      DO 16 I=1,18
      IF(IDF(I).EQ.BLK) GO TO 10
16    CONTINUE
      DO 4024 I=1,2
C     IDF(14+I)=STUDY(I)
4024  CONTINUE
      CALL PLOTID(IDF,1,9)
      GO TO 20
   10 CONTINUE
C
20    READ(5,500,END=777) (TITLA(I),I=1,18)
      IF (TITLA(1).EQ.END)  GO TO 777
      DO 4023 I=1,2
      TITLA(16+I)=STUDY(I)
4023  CONTINUE
      DO 59 I=1,NFP
      APREC(I)=0.0
59    CONTINUE
      DO 61 K=1,NFP
      DO 63 I=1,MAXI
      APREC(K)=TMPREC(I,K)+APREC(K)
63    CONTINUE
61    CONTINUE
      DO 65 I=1,NFP
      DO 67 J=1,MAXI
      TMSTFI(J,I)=(TMSTF(J,I))/VWIN
      TMOF(J,I)=TMSTFI(J,I)-TMIF(J,I)-TMBF(J,I)+TMSE(J,I)
67    CONTINUE
65    CONTINUE
      DO 322 I=1,NFP
      THSFDT=THSFD(1,I)
      PEAKS(I)=THSFD(1,I)
      PHRS(I)=1
C
      DO 320 J=2,SINDEX
      IF (THSFD(J,I).LE.THSFDT) GO TO 320
      PEAKS(I)=THSFD(J,I)
      THSFDT=THSFD(J,I)
      PHRS(I)=J
320   CONTINUE
322   CONTINUE
330   DO 332 I=1,NFP
      THSFDT=THSFD(1,I)
      DO 334 K=2,SINDEX
334   THSFDT=AMIN1(THSFDT,THSFD(K,I))
      THSFDM(I)=THSFDT
332   CONTINUE
      IF(LOOKUP.LE.0) GO TO 40
      KS=0
         READ(5,500) (TABT(I),I=1,18)
      DO 8003 I=1,2
      TABT(16+I)=STUDY(I)
8003  CONTINUE
      DO 15 J=1,50
         DO 505 K=1,10
         CALL READ(A(K))
      IF (A(K).EQ.999.0) GO TO 112
      L=K
505      CONTINUE
113   CONTINUE
C
C     STORE 10 VALUES IN TABLE
      DO 12 M=1,L
      KS=KS+1
      IF(A(M).NE.0.) NPTS=KS
      DEPEND(KS)=A(M)
12    CONTINUE
C
C     TEST FOR BLANK CARD AT END
      TEST=999.0
C     IF TEST=0,YOU HAVE THE BLANK CARD AT THE END
      IF (TEST.EQ.A(K)) GO TO 14
15    CONTINUE
14       CONTINUE
C     NPTS IS NUMBER OF POINTS,IS SET AFTER 14
         CALL READ(VARIN(1),XINC)
      DO 30 I=2,NPTS
   30 VARIN(I)= VARIN(I-1)+XINC
   40 T(1)= 0.0
      IF (CONOPT(15).NE.2) NQ=SINDEX
      DO 60 I=2,NQ
   60 T(I)= T(I-1)+DT
  606 FORMAT('1',18A4)
      IF (CONOPT(15).NE.2) GO TO 75
613   FORMAT(' WATER BASIN AREA(SQ. MI.)=',F7.1)
      IF(LOOKUP.LT.0) GO TO 75
C     CONVERT FT. OF GAUGE HEIGHT INTO SEC.FT. OF GAUGE HEIGHT.
      DO 70 I=1,NQ
```

```
   70 Q(I)= TABLE(VARIN,DEPEND,NPTS,Q(I))                                  INTG0183
C                                                                       INTG0184
   75 IFLAG= 2                                                          INTG0185
      IF(MOD(NQ,2).EQ.0) IFLAG= 1                                       INTG0186
      H= DT/24.                                                         INTG0187
C                                                                       INTG0188
      IF(LOOKUP.LT.0) GO TO 100                                         INTG0189
      WRITE(6,606) (TABT(I),I=1,18)                                     INTG0190
      WRITE(6,612) NPTS                                                 INTG0191
  612 FORMAT('ONUMBER OF POINTS IN RATING TABLE=',I6)                   INTG0192
      WRITE(6,607) (VARIN(I),I=1,NPTS)                                  INTG0193
  607 FORMAT('OINDEPENDENT VARIABLES'/(12E11.4))                        INTG0194
      WRITE(6,608) (DEPEND(I),I=1,NPTS)                                 INTG0195
  608 FORMAT(//'ODEPENDENT VARIABLES'/(12E11.4))                        INTG0196
      WRITE(6,609) (T(I),Q(I),I=1,NQ)                                   INTG0197
  609 FORMAT(//'OTIME(HRS.)',5X,'OBSERVED DISCHARGE(SEC.FT. OF GAUGE HEIINTG0198
     1GHT)' /(F9.2,F22.4))                                              INTG0199
      DO 90 I=3,5                                                       INTG0200
   90 FLDQ(I)= FLD(I+1)                                                 INTG0201
100   IF (CONOPT(15).NE.2) GO TO 25                                     INTG0202
      QMAX=Q(1)                                                         INTG0203
      PHRO=1                                                            INTG0204
      DO 161 I=1,NQ                                                     INTG0205
      IF (Q(I).LE.QMAX) GO TO 161                                       INTG0206
      QMAX=Q(I)                                                         INTG0207
      PHRO=I                                                            INTG0208
161   CONTINUE                                                          INTG0209
      QMIN= QMAX                                                        INTG0210
      DO 160 I=1,NQ                                                     INTG0211
  160 QMIN= AMIN1(QMIN,Q(I))                                            INTG0212
      QMAP= AINT(QMAX+1.)                                               INTG0213
C                                                                       INTG0214
C PRINTER PLOTS                                                         INTG0215
      NPPTS=NQ                                                          INTG0216
      INDEXM=1                                                          INTG0219
      STPTS=SINDEX                                                      INTG0220
      DO 380 K=1,STPTS                                                  INTG0221
380   SQ(K)=THSFD(K,INDEXM)                                             INTG0222
      PEAK=AINT(PEAKS(INDEXM)+1)                                        INTG0223
      IF (MOD(STPTS,2).EQ.0) IFLAG=1                                    INTG0224
      CALL INTEG(IFLAG,H,Q,SSUM,STPTS)                                  INTG0225
      SUM(2)=SSUM/(26.9*AREAI)                                          INTG0226
      CALL INTEG(IFLAG,H,SQ,SSUM,STPTS)                                 INTG0227
      SUM(1)=SSUM/(26.9*AREAI)                                          INTG0228
      KNT=1                                                             INTG0229
      M=1                                                               INTG0230
      SSUM=SUM(2)                                                       INTG0231
      DO 7 I=1,2                                                        INTG0232
         PCHAR(I)=PCHARP(I)                                             INTG0233
      PRCHAR(I)=PPCHAR(I)                                               INTG0000
7     CONTINUE                                                          INTG0234
      IF (QMAP.LT.PEAK) QMAP=PEAK                                       INTG0235
      IF (QMIN.GE.THSFDM(INDEXM)) QMIN=THSFDM(INDEXM)                   INTG0236
      CALL FLBCDE(SUM(1),6,ANSO)                                        INTG0237
      CALL FLBCDE(SUM(2),6,ANSS)                                        INTG0238
      DO 381 I=1,3                                                      INTG0239
381   SUBT(I+2)=ANSO(I)                                                 INTG0240
      DO 382 I=1,3                                                      INTG0241
382   SUBT(I+11)=ANSS(I)                                                INTG0242
      N=1                                                               INTG0243
      L=24                                                              INTG0244
      DO 53 I=1,MAXI                                                    INTG0245
      QOUT(I)=0.0                                                       INTG0246
53    CONTINUE                                                          INTG0247
      DO 57 I=1,MAXI                                                    INTG0248
      DO 55 K=N,L                                                       INTG0249
      QOUT(I)=Q(K)+QOUT(I)                                              INTG0250
55    CONTINUE                                                          INTG0251
      QOUT(I)=QOUT(I)/24.0                                              INTG0252
      N=K                                                               INTG0253
      L=N+23                                                            INTG0254
57    CONTINUE                                                          INTG0255
      APREC(3)=APREC(1)                                                 INTG0256
      GO TO 1000                                                        INTG0257
25    CONTINUE                                                          INTG0258
      PEAK=PEAKS(1)                                                     INTG0259
      QMIN=THSFDM(1)                                                    INTG0260
      DO 13 I=1,3                                                       INTG0261
      PEAK=AMAX1(PEAK,PEAKS(I))                                         INTG0262
      QMIN=AMIN1(QMIN,THSFDM(I))                                        INTG0263
13    CONTINUE                                                          INTG0264
      IF (MOD(SINDEX,2).EQ.0) IFLAG=1                                   INTG0265
      PEAK=AINT(PEAK+1)                                                 INTG0266
      QMAP=PEAK                                                         INTG0267
      DO 17 I=1,3                                                       INTG0268
      DO 19 K=1,SINDEX                                                  INTG0269
      SQ(K)=THSFD(K,I)                                                  INTG0270
19    CONTINUE                                                          INTG0271
      CALL INTEG(IFLAG,H,SQ,SSUM,SINDEX)                                INTG0272
```

```
      SUM(I)=SSUM/(26.9*AREAL)
17    CONTINUE
      M=2
      KNT=3
      QMAX=PEAKS(3)
      PHRO=PHRS(3)
      SSUM=SUM(3)
      DO 11 I=1,3
      PCHAR(I)=PCHARF(I)
      PRCHAR(I)=PFCHAR(I)
11    CONTINUE
      DO 33 K=1,SINDFX
      Q(K)=THSFD(K,1)
 33   CONTINUE
      DO 3 I=1,18
      SUBT(I)=FLDT(I)
      FLDY(I)=FLDQ(I)
3     CONTINUE
1000  IF (IPLOT.NE. 0) GO TO 1001
      CALL PL360(NPPTS,TEMP,T,T(1),T(NQ),SUBT,Q,QMIN,QMAP,FLDY,
     1TITLA,-214)
      CALL PL360(STPTS,TEMP,T,T(1),T(NQ),SUBT,SQ,QMIN,
     1QMAP,FLDY,TITLA,226)
C
C SC4020 PLOTS
1001  IF (IPLOT.EQ.0) GO TO 391
      CALL PL4020(1,1,PCHAR(1),SINDEX,T,Q,T(1),T(NQ),QMIN,QMAP,SUBT,
     1FLDY,TITLA,IERR)
      CALL PL360(SINDEX,TEMP,T,T(1),T(NQ),SUBT,Q,QMIN,QMAP,FLDY,TITLA,
     *PRCHAR(1))
      IF (IERR.EQ.1) GO TO 185
191   IF (IERR.EQ.2) GO TO 187
      WRITE (6,715)
715   FORMAT (' OFF SCALE PLOT POINTS WERE ENCOUNTERED')
      GO TO 20
187   WRITE (6,717)
717   FORMAT(' SUBROUTINES UNABLE TO CONSTRUCT READABLE GRIDS')
185   DO 21 I=M,KNT
      DO 23 K=1,SINDEX
      SQ(K)=THSFD(K,I)
23    CONTINUE
      IF (CONOPT(15).EQ.2)  PCHAR(1)=PCHAR(2)
      IF  (CONOPT(15) .EQ.2)  PRCHAR(1)=PPCHAR(2)
      CALL PL4020(2,1,PCHAR(I),SINDEX,T,SQ,T(1),T(NQ),QMIN,
     1QMAP,SUBT,FLDY,TITLA,IERR)
      CALL PL360(SINDEX,TEMP,T,T(1),T(NQ),SUBT,SQ,QMIN,QMAP,FLDY,TITLA,
     *PRCHAR(I))
21    CONTINUE
      IF (IERR.NE.1) GO TO 191
690   FORMAT(' ',18A4)
C OBTAIN THE DIFFERENCE BETWEEN THE OBSERVED AND SIMULATED MAX FOR DAY
C
391   DIFFS=PEAKS(1)-QMAX
      DIFFSA=DIFFS
      IF(DIFFS.LT.0.0) DIFFSA=-DIFFS
      DIFFMP=(DIFFSA/QMAX)*100.0
C
C OBTAIN THE DIFFERENCE BETWEEN THE OBSERVED AND SIMULATED PEAKS
      DIFFP=PHRS(1)-PHRO
      DIFFPA=DIFFP
      IF(DIFFP.LT.0) DIFFPA=-DIFFP
      DIFFPP=(DIFFPA/PHRO)*100
C
C
C COMPUTE DIFFERENCE AND %DIFFERENCE IN RUNOFF
      DIFFR=SUM(1)-SSUM
      DIFFRA=DIFFR
      IF(DIFFR.LT.0.0) DIFFRA=-DIFFR
      DIFFRP=(DIFFRA/SSUM)*100.0
C
C
C     COMPUTE DIFFERENCE BETWEEN FORCASTED&WORSTCASE PRECIP
      DIFFPR=APREC(1)-APREC(3)
      DIFPP=DIFFPR/APREC(1)*100
      WRITE (6,700) (TITLE(I),I=1,18)
      IF (CONOPT(15).EQ.2) GO TO 729
700   FORMAT('1',18A4)
      WRITE (6,710) MSBOIC
710   FORMAT(63X,'TABLE',1X,'ONE',3X,'FORECAST RUN',/,64X,A4,10X,'WORST
     1CASE',10X,'NO PRECIP',10X,'FORECAST',/)
      DO 71 I=1,MAXI
      WRITE (6,711) MPDAY(I),(TMPREC(I,K),K=1,3),(TMOF(I,K),K=1,3),
     1(TMIF(I,K),K=1,3),(TMBF(I,K),K=1,3),(STMROS(I,K),K=1,3)
711   FORMAT(64X,I2,2X,'PRECIP',8X,F5.3,2(15X,F5.3),/,68X,'SUR R/O',7..,
     1F5.3,2(15X,F5.3),/,68X,'INT FL',8X,F5.3,2(15X,F5.3),/,68X,
     2'BASE FL',7X,F5.3,2(15X,F5.3),/,68X,'STM R/O',3X,F7.1,2(15X,F7.1),
     3//)
71    CONTINUE
      WRITE (6,700) (TITLE(I),I=1,18)
```

```
      WRITE (6,712)                                                     INTG0356
712   FORMAT(63X,'TABLE',1X,'TWO',3X,'FORECAST RUN',/,43X,'WORST CASE', INTG0357
     110X,'NO PRECIP',10X,'FORECAST',10X,'DIFF  ',8X,'%DIFF',/)         INTG0358
      WRITE (6,713) (PEAKS(I),I=1,3),DIFFS,DIFFMP                       INTG0359
713   FORMAT(31X,'PEAK (CFS)',3X,F7.1,11X,F7.1,11X,F7.1,8X,F8.2,8X,F5.2,INTG0360
     1//)                                                               INTG0361
      WRITE(6,714) (PHRS(I),I=1,3),DIFFP,DIFFPP                         INTG0362
714   FORMAT(31X,'PEAK (HR)',6X,I3,15X,I3,15X,I3,12X,I3,10X,F5.2,//)    INTG0363
      WRITE (6,719) (SUM(I),I=1,3),DIFFR,DIFFRP                         INTG0364
719   FORMAT(31X,'R/O (IN)',9X,F5.2,12X,F5.2,12X,F5.2,12X,F5.2,10X,F5.2,INTG0365
     1//)                                                               INTG0366
      WRITE(6,721) (APREC(I),I=1,3),DIFFPR,DIFPP                        INTG0367
721   FORMAT(31X,'PRECIP (IN)',2X,F7.3,11X,F7.3,11X,F7.3,8X,F8.3,8X,    INTG0368
     1F5.2)                                                             INTG0369
      WRITE (6,700) (TITLE(I),I=1,18)                                   INTG0000
      WRITE(6,723) MSBDIC                                               INTG0371
723   FORMAT(63X,'TABLE',1X,'THREE',3X,'FORECAST RUN',/,64X,A4,10X,     INTG0372
     1'WORST CASE',10X,'NO PRECIP',10X,'FORECAST',/)                    INTG0373
      DO 725 I=1,MAXI                                                   INTG0374
      WRITE(6,727) MPDAY(I),(EMIFS(I,K),K=1,3),(EMUZS(I,K),K=1,3),      INTG0375
     1(EMLZS(I,K),K=1,3),(EMGWS(I,K),K=1,3),(EMBFNX(I,K),K=1,3),        INTG0376
     2(EMSIAM(I,K),K=1,3),(EMUZC(I,K),K=1,3)                            INTG0377
727   FORMAT(64X,I2,2X,'IFS',11X,F5.2,2(15X,F5.2),/,68X,'UZS',11X,F5.2, INTG0378
     12(15X,F5.2)/,68X,'LZS',11X,F5.2,2(15X,F5.2),/,68X,'GWS',11X,F5.2, INTG0379
     22(15X,F5.2),/,68X,'BFNX',11X,F5.2,2(15X,F5.2),/,68X,'SIAM'10X,    INTG0380
     3F5.2,2(15X,F5.2),/,68X,'UZC ',11X,F5.2,2(15X,F5.2),/)             INTG0381
725   CONTINUE                                                          INTG0382
      GO TO 751                                                         INTG0383
729   CONTINUE                                                          INTG0384
      WRITE (6,731) MSBDIC                                              INTG0385
731   FORMAT(63X,'TABLE',1X,'ONE',10X,'PAST RUN',/,64X,A4,10X,'OBSERVED'INTG0386
     1,10X,'SIMULATED',//)                                              INTG0387
      DO 85 I=1,MAXI                                                    INTG0388
      WRITE(6,735) MPDAY(I),TMPREC(I,1),TMPREC(I,1),TMOF(I,1),TMIF(I,1),INTG0389
     1TMBF(I,1),QOUT(I),STMROS(I,1)                                     INTG0390
735   FORMAT(66X,I2,'PRECIP',8X,F5.3,15X,F5.3,/,68X,'SUR R/O',27X,F5.3, INTG0391
     1/,68X,'INT FL',28X,F5.3,/,68X,'BASE FL',27X,F5.3,/,68X,'STM R/O', INTG0392
     27X,F7.1,10X,F7.1,//)                                              INTG0393
85    CONTINUE                                                          INTG0394
      WRITE (6,700) (TITLE(I),I=1,18)                                   INTG0000
      WRITE (6,737)                                                     INTG0396
737   FORMAT(63X,'TABLE',1X,'TWO',10X,'PAST RUN',/,46X,'OBSERVED',2X,   INTG0397
     1'SIMULATED',3X,'DIFF',3X,'% DIFF',/)                              INTG0398
      WRITE (6,739) QMAX,PEAKS(1),DIFFS,DIFFMP                          INTG0399
739   FORMAT(11X,'PEAK (CFS)',23X,2(F7.1,3X),F8.2,1X,F5.1,/)            INTG0400
      WRITE(6,741) PHRO,PHRS(1),DIFFP,DIFFPP                            INTG0401
741   FORMAT(11X,'PEAK (HR)',24X,I3,9X,I3,9X,I3,2X,F5.1,/)              INTG0402
      WRITE (6,743) SUM(1),SUM(2),DIFFR,DIFFRP                          INTG0403
743   FORMAT(11X,'R/O (IN)',27X,F5.2,5X,F5.2,6X,F5.2,1X,F5.1,/)         INTG0404
      WRITE (6,745) APREC(1),APREC(1)                                   INTG0405
745   FORMAT(11X,'PREPCIP IN',23X,F7.3,3X,F7.3)                         INTG0406
      WRITE (6,700) (TITLE(I),I=1,18)                                   INTG0000
      WRITE (6,747) MSBDIC                                              INTG0407
747   FORMAT(63X,'TABLE',1X,'THREE',2X,'PAST RUN',/,64X,A4,10X,'SIMULATEINTG0408
     1D')                                                               INTG0409
      DO 87 I=1,MAXI                                                    INTG0410
      WRITE(6,749) MPDAY(I),EMIFS(I,1),EMUZS(I,1),EMLZS(I,1),EMGWS(I,1),INTG0411
     1EMBFNX(I,1),EMSIAM(I,1),EMUZC(I,1)                                INTG0412
749   FORMAT(66X,I2,2X,'IFS ',10X,F5.2,/,70X,'UZS ',10X,F5.2,/,70X,     INTG0413
     1'LZS ',10X,F 5.2,/,70X,'GWS ',10X,F5.2,/,70X,'BFNX',10X,F5.2,/,   INTG0414
     270X,'SIAM',10X,F5.2,/,70X,'UZC ',10X,F5.2,//)                     INTG0415
87    CONTINUE                                                          INTG0416
751   CONTINUE                                                          INTG0417
777   CONTINUE                                                          INTG0418
      RETURN                                                            INTG0419
C                                                                       INTG0420
  603 FORMAT('OPOINT NUMBER',I5,' IS OUTSIDE PLOT LIMITS')              INTG0421
112   L=K-1                                                             INTG0422
      GO TO 113                                                         INTG0423
      END                                                               INTG0424
      SUBROUTINE INTEG(IFLAG,DT,Q,SUM,NQ)                               INTG0425
C                                                                       INTG0426
      DIMENSION  Q(1)                                                   INTG0427
      SUM= 0.0                                                          INTG0428
      IF(IFLAG.EQ.2) GO TO 80                                           INTG0429
      LU= NQ-1                                                          INTG0430
      DO 40 I=2,LU                                                      INTG0431
   40 SUM= SUM+2.*Q(I)                                                  INTG0432
      SUM=(SUM+Q(1)+Q(NQ))*DT*.5                                        INTG0433
      RETURN                                                            INTG0434
C                                                                       INTG0435
   80 LU= NQ-3                                                          INTG0436
      DO 100 I=2,LU,2                                                   INTG0437
  100 SUM= SUM+4.*Q(I)+2.*Q(I+1)                                        INTG0438
      SUM= DT/3.*(Q(1)+SUM+4.*Q(NQ-1)+Q(NQ))                            INTG0439
      RETURN                                                            INTG0440
      END                                                               INTG0441
      FUNCTION TABLE(X,Y,N,VALX)                                        INTG0442
      DIMENSION X(1),Y(1)                                               INTG0443
```

```
C
C     SUBROUTINE TO PERFORM A TABLE LOOKUP AND LINEAR INTERPOLATION
C
      YINT(X1,X2,Y1,Y2) = (Y1-Y2)*(VALX-X2)/(X1-X2) + Y2
C
C     TEST FOR VALX OUTSIDE RANGE OF X TABLE
      IF(VALX.GT.X(1)) GO TO 10
      I = 2
      GO TO 100
   10 CONTINUE
      IF(VALX.LT.X(N)) GO TO 20
      I = N
      GO TO 100
C
C     VALX WITHIN RANGE OF X TABLE
   20 CONTINUE
      J = N/2
      I = 1
      IF(VALX.GT.X(J)) I=J
   30 CONTINUE
      IF(VALX.LT.X(I)) GO TO 100
      I = I+1
      GO TO 30
C
C     FIND INTERPOLATED VALUE
  100 CONTINUE
      TABLE = YINT(X(I-1),X(I),Y(I-1),Y(I))
      RETURN
      END
```

```
      SUBROUTINE LABELS(BCDX,BCDY,IDF,IERR)
      COMMON/GSPD/WORK   ,GSP1   ,UNITIN ,I2250  ,IATN   ,OAREA  ,
     1       NULLV  ,IGDSC  ,IPTNOW,MAXPT ,IPT    ,ISYM   ,RWORK1,
     2       IGDS4  ,IGDS5  ,IGDS6  ,IOSP   ,RU     ,HSPB   ,HSPL   ,
     3       VSPB   ,VSPL   ,NCUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4       XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5       YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6       AY     ,BY     ,YLPOS  ,
     7       LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8       FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      LOGICAL*1 GSPIN, DUM(2)
      INTEGER*4 PTLID(3)/'PLOT',' NO ','    '/
      INTEGER*4 EDVAL(3),ZERO(3)/'    ','  0 ','    '/
      INTEGER*4 IDF(18),BCDX(18),BCDY(18),BCDY2(72)
      REAL*8 DYDVAL,WORK(36)
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      INTEGER*4 UNITIN,OAREA,NULLV(1),GSP1
      LOGICAL*1 FA,FB,FC,GMODE1
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      EQUIVALENCE (RWORK1,RN),(ISYM, IRNO)
C   CREATE PLOT TITLE LABEL
      ICNT=IFIX(XSIZE/HSPB)+5
      RN=XLL-5.*HSPB
      CALL PTEXT(IGDSC,IDF,ICNT,NULLV,NULLV,1,RN,YUR+5.*VSPB)
C   CREATE X AXIS LABEL ELEMENTS
      CALL PTEXT(IGDSC,BCDX,ICNT,NULLV,NULLV,1,RN,YLL-3.5*VSPB)
C   CREATE Y AXIS LABEL ELEMENTS
       CALL PLOTL(BCDY,BCDY2,72,IRNO)
      RN=YUR + 6.*VSPB
      IRNO=MINO(IRNO,IFIX(YSIZE/VSPB)+5)
       DO 40 I = 1,IRNO
       RN = RN - VSPB
       CALL PTEXT(IGDSC,BCDY2(I),1,NULLV,NULLV,1,0.0,RN)
40     CONTINUE
      IF(IERR .EQ. 2)GO TO 70
C   CONVERT Y DIVISION VALUES TO EBCDIC FORMAT AND CREATE
C   THE TEXT ELEMENTS NECESSARY FOR DISPLAYING
      DO 50 I=1,NYD
      DYDVAL=YDVAL(I)
      CALL TC3(DYDVAL,11,EDVAL)
   50 CALL PTEXT(IGDSC,EDVAL,11,NULLV,NULLV,1,XLL-13.*HSPB,YTIC(I))
C   CONVERT X DIVISION VALUES TO EBCDIC FORMAT AND CREATE
C   THE TEXT ELEMENTS NECESSARY FOR DISPLAYING
      DO 60 I=1,NXD
      DYDVAL=XDVAL(I)
      CALL TC3(DYDVAL,9,EDVAL)
       IF(XDVAL(I) .NE. 0.0) GO TO 60
       EDVAL(1) = ZERO(1)
       EDVAL(2) = ZERO(2)
       EDVAL(3) = ZERO(3)
   60 CALL PTEXT(IGDSC,EDVAL,9,NULLV,NULLV,1,XTIC(I)-6.*HSPB,
     1     YLL-1.5*VSPB)
C   CREATE PLOT ID
   70 CONTINUE
      RN=XLL+(XSIZE/2.-3.*HSPB)
      RETURN
      END
```

```
H016     TITLE '    '
*
H016CG   START
*
*        WRITTEN BY C G HOOKS            IBM HUNTSVILLE
*                            CALL PL360 (N, A, X, XMIN, XMAX, XLABEL,
*                                 Y, YMIN, YMAX, YLABEL, HEAD, NSYM)
*
*                                PL360 PLOTS N POINTS USING THE COORDI-
*                                   NATES FROM ARRAYS X AND Y. THE
*                                   SCALES ARE SET USING THE MAXIMUM AND
*                                   MINIMUM VALUES XMIN, XMAX, YMIN,
*                                   YMAX. THE GRAPH IS LABLED AT THE TOP
*                                   WITH 18 WORDS FROM HEAD, WITH 13
*                                   WORDS FROM YLABLE ON THE LEFT, AND
*                                   18 WORDS FROM XLABLE AT THE BOTTOM.
*                                   ARRAY A IS SUPPLIED FROM THE MAIN
*                                   PROGRAM FOR TEMPORARY STORAGE. ARRAY
*                                   MUST HAVE A LENGTH OF AT LEAST 1540
*                                   FULL WORDS. NSYM IS THE DECIMAL
*                                   EQUIVALENT OF THE PLOT CHARACTER.
*                                   THE GRAPH WILL BE PLOTTED IF NSYM IS
*                                   POSITIVE. IF NSYM IS NEGATIVE THE
*                                   POINTS WILL BE STORED BUT NOT PLOTED
*
         EXTRN PR360
         ENTRY PL360
         USING *,15
PL360    BC    15,GO
         DC    X'06'
         DC    CL5'PL360'
GO       STM   14,12,12(13)
         LR    0,1
         DS    0H
         CNOP  4,8
         BAL   1,*+76
         DS    CL72
         XC    0(72,1),0(1)
         ST    1,8(13)
         ST    13,4(1)
         LR    13,1
BEGIN    BALR  9,0
         USING *,9
         LR    1,0
         MVC   N(48),0(1)              STORE ADDRESSES OF ARGUMENTS
         L     10,A                    LOAD R10 ADDRESS OF ARRAY
         ST    10,AARRAY            STORE ADDRESS OF ARRAY FOR PRINTING
         L     11,NSYM
         L     3,0(0,11)            LOAD R3 WITH NSYM
         LPR   3,3
*                                      TEST FOR MODE OF PLOT
         SR    0,0
         ST    0,IXLOG
         ST    0,IYLOG
         C     3,C1000
         BC    4,POS                   MODE = 1  X-LINEAR   Y-LINEAR
T2000    C     3,C2000
         BC    2,T3000
         S     3,C1000                 MODE = 2  X-LOG      Y-LINEAR
         MVC   IXLOG(4),ONE
         BC    15,POS
T3000    C     3,C3000
         BC    2,T4000
         S     3,C2000                 MODE = 3  X-LINEAR   Y-LOG
         MVC   IYLOG(4),ONE
         BC    15,POS
T4000    S     3,C3000                 MODE = 4  X-LOG      Y-LOG
         MVC   IXLOG(4),ONE
         MVC   IYLOG(4),ONE
POS      BCR   0,0
         L     11,HEAD                 R11 ADDRESS OF HEADNG
         MVC   20(72,10),0(11)         STORE HEADING RIGHT ADJUSTED 20
         L     11,XLABEL               R11 ADDRESS OF XLABLE
         A     10,XLAB                 ADJUST ADDRESS OF ARRAY
         MVC   0(72,10),0(11)          STORE XLABLE 72 CHARACTERS
         L     10,A                    RESTORE ADDRESS OF ARRAY
         L     11,YLABEL               R11 ADDRESS OF YLABLE
         L     4,CTLY
LOOPY    IC    0,0(0,11)
         STC   0,226(0,10)             STORE CHARACTER    226 = 2 X 112 + 2
         A     11,ONE                  INCREMENT
         A     10,ONEHUN               INCREMENT
         BCT   4,LOOPY                 BRANCH ON NOT FINISHED
         L     10,A                    RESTORE ADDRESS OF ARRAY
         L     11,XMAX                 XMAX ADDRESS
         L     7,XMIN                  XMIN ADDRESS
         L     12,YMAX                 YMAX ADDRESS
         L     8,YMIN                  YMIN ADDPESS
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
*
TYMODE CLC   IYLOG(4),ONE          TEST FOR LOG SCALE Y
       BC    7,YLINSC
       LA    1,YMAX                SET UP LOG SCALE Y
       L     15,ALG10              LOG YMAX
       BALR  14,15
       BC    0,2                   ISN
       LER   6,0
       LA    1,YMIN
       L     15,ALG10              LOG YMIN
       BALR  14,15
       BC    0,4                   ISN
       STE   0,LYMIN
       SER   6,0
       LE    4,CTY
       DER   4,6
       STE   4,TEMPY
       LE    0,0(12)               CALCULATE Y SCALE LOG
       DE    0,0(8)
       STE   0,ROOT
       LA    1,AROOT
       L     15,AFRXPR             FIFTH ROOT OF YMAX/YMIN
       BALR  14,15
       BC    0,6                   ISN
       STE   0,YROOT5
       LE    4,0(12)               YMAX
       L     2,SIXFX
*
YLSCLE STE   4,228(10)             STORE Y SCALE LOG
       DER   4,0                   / FIFTH ROOT
       A     10,CT1120
       BCT   2,YLSCLE
       BC    15,TXMODE
YLINSC LE    0,0(0,12)             CONSTANT FOR Y CALCULATION
       SE    0,0(0,8)
       LE    4,CTY
       DER   4,0
       STE   4,TEMPY
       LE    4,FIVE                CALCULATE AND STORE Y SCALE
       DER   0,4
       LE    4,0(0,12)
       L     2,SIXFX
YSCALE STE   4,228(0,10)           227 = 2 X 112 + 4
       SER   4,0
       A     10,CT1120
       BCT   2,YSCALE
*
TXMODE CLC   IXLOG(4),ONE          TEST FOR LOG SCALE X
       BC    7,XLINSC
       LA    1,XMAX                SET UP LOG SCALE X
       L     15,ALG10              LOG XMAX
       BALR  14,15
       BC    0,8                   ISN
       LER   6,0
       LA    1,XMIN
       L     15,ALG10              LOG XMIN
       BALR  14,15
       BC    0,10                  ISN
       STE   0,LXMIN
       SER   6,0
       LE    4,CTX
       DER   4,6
       STE   4,TEMPX
       LE    0,0(11)               CALCULATE X SCALE LOG
       DE    0,0(7)
       STE   0,ROOT
       LA    1,AROOT
       L     15,AFRXPR             FIFTH ROOT OF XMAX/XMIN
       BALR  14,15
       BC    0,12                  ISN
       STE   0,XROOT5
       LE    4,0(7)                XMIN
       L     10,A
       A     10,XSCAL
       L     2,SIXFX
*
XLSCLE STE   4,0(10)
       MER   4,0                   * FIFTH ROOT
       A     10,FOUR
       BCT   2,XLSCLE
       BC    15,STOPTS
XLINSC LE    0,0(0,11)             CONSTANT FOR X CALCULATION
       SE    0,0(0,7)
       LE    4,CTX
       DER   4,0
       STE   4,TEMPX
       L     10,A                  CALCULATE AND STORE X SCALE
       A     10,XSCAL
       L     2,SIXFX
```

```
         LE    4,FIVE
         DER   0,4
         LE    4,0(0,7)
XSCALE   STE   4,0(0,10)
         AER   4,0
         A     10,FOUR
         BCT   2,XSCALE
*
STOPTS   L     14,N                      R14 ADDRESS OF NCHAR
         L     4,0(0,14)                 STORE NCHAR IN R4
         L     5,X                       R5  ADDRESS OF X
         L     6,Y                       R6  ADDRESS OF Y
         SDR   2,2                       R2 = 0
         SDR   0,0                       R0 = 0
         L     10,A
LOOPXY   LE    6,0(0,6)                  CALCULATE POSITION FOR NCHAR
         CE    6,0(0,12)
         BC    2,ERROR
         SE    6,0(0,8)
         BC    4,ERROR
         LTER  6,6
         BC    8,YLIN
*
         CLC   IYLOG(4),ONE
         BC    7,YLIN
         AE    6,0(0,8)
         STE   6,DDY
         LA    1,ADDY
         L     15,ALG10                  LOG Y
         BALR  14,15
         BC    0,14                      ISN
         LER   6,0
         SE    6,LYMIN
YLIN     ME    6,TEMPY
         AE    6,PT5
         AW    6,FZERO
         AD    6,FZERO
         ME    6,E112
         LE    2,0(0,5)
         CE    2,0(0,11)
         BC    2,ERROR
         SE    2,0(0,7)
         BC    4,ERROR
         LTER  2,2
         BC    8,XLIN
*
         CLC   IXLOG(4),ONE
         BC    7,XLIN
         AE    2,0(0,7)
         STE   2,DDX
         LA    1,ADDX
         L     15,ALG10                  LOG X
         BALR  14,15
         BC    0,16                      ISN
         LER   2,0
         SE    2,LXMIN
XLIN     ME    2,TEMPX
         LE    4,CTLOW
         AE    2,PT5
         SER   4,6
         AER   2,4
         AW    2,FZERO
         STD   2,TEMPW
         L     14,TEMPW+4
         STC   3,232(14,10)              238 = 2 X 112 + 8
         A     5,FOUR                    INCREMENT X ADDRESS
         A     6,FOUR                    INCREMENT Y ADDRESS
         BCT   4,LOOPXY
         MVI   0(10),X'F1'
*
*        WRITE OUT GRAPH
         L     11,NSYM
         L     3,0(0,11)
         LTR   3,3
         BC    2,PRT
NOPRT    BCR   0,0
         DS    0H
         L     13,4(13)
         RETURN  (14,12),T,RC=0
PRT      LA    1,AARRAY
         L     15,APR360
         BALR  14,15            TRANSFER TO PR360 ROUTINE
         BC    0,18                      ISN
         BC    15,NOPRT
ERROR    L     14,N
         L     0,0(0,14)
         SR    0,4
         A     0,ONE
         LNR   0,0
```

```
         ST    0,0(0,14)
         BC    15,NOPRT
AARRAY   DS    F
APR360   DC    A(PR360)
XLAB     DC    F'5956'
ONE      DC    F'1'
ONEHUN   DC    F'112'
CTY      DC    E'50'
PT5      DC    E'.5'
E112     DC    E'112'
CTX      DC    E'100'
CTLY     DC    F'50'
FOUR     DC    F'4'
TEMPX    DS    F
TEMPY    DS    F
TEMPW    DS    D
FZERO    DC    X'4E00000000000000'
FIVE     DC    E'5'
SIXFX    DC    F'6'
CT1120   DC    F'1120'
XSCAL    DC    F'6056'            6056 = 112 X 54 + 8
CTLOW    DC    E'5600'            5600 = 112 X 50
C1000    DC    F'1000'
C2000    DC    F'2000'
C3000    DC    F'3000'
IXLOG    DS    F
IYLOG    DS    F
LXMIN    DS    F
LYMIN    DS    F
AFRXPR   DC    V(FRXPR#)
ALG10    DC    V(ALOG10)
N        DS    F                    ADDRESSES OF ARGUMENTS
A        DS    F
X        DS    F
XMIN     DS    F
XMAX     DS    F
XLABEL   DS    F
Y        DS    F
YMIN     DS    F
YMAX     DS    F
YLABEL   DS    F
HEAD     DS    F
NSYM     DS    F
ROOT     DS    F                    NUMBER TO TAKE FIFTH ROOT OF
AROOT    DC    A(ROOT)
         DC    X'80'
         DC    AL3(FIFTH)
FIFTH    DC    E'.2'
XROOT5   DS    F
DDX      DS    F
ADDX     DC    X'80'
         DC    AL3(DDX)
YROOT5   DS    F
DDY      DS    F
ADDY     DC    X'80'
         DC    AL3(DDY)
         END
```

```
      SUBROUTINE PL4020(NPLOT,MODE,NCHAR,NP,X,Y,XMIN,XMAX,
     1                  YMIN,YMAX,BCDX,BCDY,IDF,IERR)
C    THIS IS THE MAIN ROUTINE FOR GENERATING 2250 PLOTS
C
C    INPUT PARAMETER DEFINITION:
C
C       NPLOT      - CONTROLS NUMBER OF CURVES PER DISPLAY
C                    (NPLOT=1 START NEW DISPLAY)
C                    (NPLOT=2 SUPERIMPOSE CURVE ON PREVIOUS DISPLAY)
C       MODE       - TYPE OF GRID
C                    = 1 X-LINEAR AND Y-LINEAR
C                    = 2 X-LOG    AND Y-LINEAR
C                    = 3 X-LINEAR AND Y-LOG
C                    = 4 X-LOG    AND Y-LOG
C       NCHAR      - SELECTS PLOT SYMBOL TO BE USED
C       NP         - NUMBER OF POINTS TO BE PLOTTED
C       X,Y        - NAMES OF ARRAYS CONTAINING THE X AND Y COORDINATES
C       XMIN,XMAX  - MINIMUM AND MAXIMUM VALUES FOR X COORDINATES
C       YMIN,YMAX  - MINIMUM AND MAXIMUM VALUES FOR Y COORDINATES
C       BCDX,BCDY  - CHARACTER LABELS FOR X AND Y AXES (72 CHAR MAX)
C       IDF        - NAME OF 72 CHARACTER HEADING FOR EACH DISPLAY
C       IERR       - ERROR INDICATOR.  NOT SET BY THIS ROUTINE
C                    SINCE ERROR MESSAGES ARE DISPLAYED ON THE
C                    2250 SCREEN.  STANDARD MEANINGS ARE GIVEN BELOW.
C                     = 1 NORMAL RETURN
C                     = 2 UNABLE TO CONSTRUCT READABLE GRAPH
C                     = 3 OFF-SCALE PLOT POINTS ENCOUNTERED
C
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     1       NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2       IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3       VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4       XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5       YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6       AY     ,BY     ,YLPOS  ,
     7       LOGXSW,LOGYSW,NPTF  ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8       FC     ,GMODE1,GSPIN ,DUM    ,IRECAL
      LOGICAL*1 GSPIN, DUM(2)
      REAL*8 ERR1(6)/'NO POINT','S SPECIF','IED FOR ','1 OR MOR',
     1       'E CURVES','        '/
      REAL*8 ERR2(6)/'UNABLE T','O CONSTR','UCT READ','ABLE GRA',
     1       'PH      ','        '/
      REAL*8 PLTID(2)/'<<< ERRO','RS >>>  '/
      DIMENSION A(1)
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      REAL*4 LGXMAX,LGXMIN,LGYMAX,LGYMIN
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      LOGICAL*1 FA,FB,FC,GMODE1
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      DIMENSION X(NP),Y(NP)
      REAL*8 BCDX(9),BCDY(9),IDF(9),WORK(36)
      IERR = 1
      GO TO(10,200),NPLOT
   10 CONTINUE
      CALL SHIFT1
      LOGXSW = .FALSE.
      LOGYSW = .FALSE.
      FA=.FALSE.
      IPTNOW=0
      ERRHD=.TRUE.
      NPTF=.FALSE.
      NRGF=.FALSE.
      WRITE(NOUT,1020)IDSP,(IDF(I),I=1,9),NPLOT,MODE,NCHAR,NP,
     1      XMIN,XMAX,YMIN,YMAX
   13 CALL SGRID
      IF(XMAX .LE. XMIN .OR. YMAX .LE. YMIN) GO TO 100
      GO TO(20,30,40,50),MODE
C
C    MODE = 1    X-LINEAR  Y-LINEAR
C
   20 CONTINUE
      AX = XSIZE/(XMAX-XMIN)
      BX = XLL - AX*XMIN
      AY = YSIZE/(YMAX-YMIN)
      BY = YLL - AY*YMIN
      XINC = (XMAX - XMIN)/(FLOAT(NXD) - 1.0)
      YINC = (YMAX - YMIN)/(FLOAT(NYD) - 1.0)
      GO TO 60
C
C    MODE = 2    X-LOG      Y-LINEAR
C
   30 CONTINUE
      IF(XMIN .LE. 0.0) GO TO 100
      LGXMIN = ALOG10(XMIN)
      LGXMAX = ALOG10(XMAX)
      LOGXSW = .TRUE.
      AX = XSIZE/(LGXMAX - LGXMIN)
      BX = XLL - AX*LGXMIN
      AY = YSIZE/(YMAX - YMIN)
```

```
      BY = YLL - AY*YMIN
      XINC = (XMAX/XMIN)**(1.0/(FLOAT(NXD)-1.0))
      YINC = (YMAX-YMIN)/(FLOAT(NYD)-1.0)
      GO TO 60
C
C   MODE = 3    X-LINEAR Y-LOG
C
   40 CONTINUE
      IF(YMIN .LE. 0.0) GO TO 100
      LGYMAX = ALOG10(YMAX)
      LGYMIN = ALOG10(YMIN)
      LOGYSW = .TRUE.
      AX = XSIZE/(XMAX-XMIN)
      BX = XLL - AX*XMIN
      AY = YSIZE/(LGYMAX-LGYMIN)
      BY = YLL - AY*LGYMIN
      XINC = (XMAX-XMIN)/(FLOAT(NXD)-1.0)
      YINC = (YMAX/YMIN)**(1.0/(FLOAT(NYD)-1.0))
      GO TO 60
C
C   MODE = 4    X-LOG     Y-LOG
C
   50 CONTINUE
      IF(XMIN .LE. 0.0)GO TO 100
      IF(YMIN .LE. 0)GO TO 100
      LGXMIN = ALOG10(XMIN)
      LGXMAX = ALOG10(XMAX)
      LGYMAX = ALOG10(YMAX)
      LGYMIN = ALOG10(YMIN)
      LOGXSW = .TRUE.
      LOGYSW = .TRUE.
      AX = XSIZE/(LGXMAX-LGXMIN)
      BX = XLL - AX*LGXMIN
      AY = YSIZE/(LGYMAX-LGYMIN)
      BY = YLL - AY*LGYMIN
      XINC = (XMAX/XMIN)**(1.0/(FLOAT(NXD)-1.0))
      YINC = (YMAX/YMIN)**(1.0/(FLOAT(NYD)-1.0))
C
   60 CONTINUE
      XDVAL(1) = XMIN
      YDVAL(1) = YMIN
      DO 72 I=2,NXD
      IF(.NOT.LOGXSW)GO TO 70
      XDVAL(I)=XDVAL(I-1)*XINC
      GO TO 72
   70 XDVAL(I) = XDVAL(I-1) + XINC
   72 CONTINUE
      DO 82 I=2,NYD
      IF(.NOT.LOGYSW)GO TO 80
      YDVAL(I)=YDVAL(I-1)*YINC
      GO TO 82
   80 YDVAL(I) = YDVAL(I-1) + YINC
   82 CONTINUE
      IF(NP)210,210,90
   90 CALL POINTS(X,Y,NP,NCHAR,IERR,XMIN,XMAX,YMIN,YMAX,PLTID)
   81 CALL LABELS(BCDX,BCDY,IDF,IERR)
C   ENTER PLOT TITLE IN DICTIONARY
      CALL DICTRY(IDF)
      GO TO 205
C
  100 CONTINUE
      CALL PTEXT(IGDSC,ERR2,48,NULLV,NULLV,1,0.0,YLL-10.*VSPB)
      WRITE(NOUT,1030)(ERR2(I),I=1,6)
  103 NRGF=.TRUE.
      IERR=2
      GO TO 211
C
C   CONTROL PASSES TO STMT 200 WHEN NPLOT EQ 2
C
  200 CONTINUE
      WRITE(NOUT,1010)NPLOT,NCHAR,NP
      IF(NRGF)GO TO 230
      IF(NP .LE. 0)GO TO 210
      CALL POINTS(X,Y,NP,NCHAR,IERR,XMIN,XMAX,YMIN,YMAX,PLTID)
  205 CONTINUE
      IF(GMODE1)GO TO 230
      CALL EXEC(IGDSC)
      CALL ATTEN
      GO TO 230
  210 WRITE(NOUT,1030)(ERR1(I),I=1,6)
      IF(NPTF)GO TO 212
      NPTF=.TRUE.
      CALL PTEXT(IGDSC,ERR1,48,NULLV,NULLV,1,0.0,YLL-8.5*VSPB)
  211 IF(ERRHD)CALL PTEXT(IGDSC,PLTID,14,NULLV,NULLV,1,0.,YLL-5.5*VSPB)
      ERRHD=.FALSE.
  212 GO TO(81,230),NPLOT
  230 CONTINUE
      IERR=1
      RETURN
```

```
C
C     DUMMY ENTRY FOR TX4020
C
      ENTRY TX4020(NPLOT,A,NCHAR,NLINE,NCOL,IER)
      WRITE(NOUT,2000)
      RETURN
 1010 FORMAT(' PL4020:  NPLOT=',I1,'     NCHAR=',I2,'     NP=',I6)
 1020 FORMAT(' PL4020:  PLOT=',I5,3X,9A8/10X,'NPLOT=',I1,'    MODE=',
     1      I1,'    NCHAR=',I2,'    NP=',I5,'    XMIN=',E12.4,
     2      '    XMAX=',E12.4,'    YMIN=',E12.4,'    YMAX=',E12.4)
 1030 FORMAT(' ***',6A8)
 2000 FORMAT(' **** TX4020 ENTRY NOT IMPLEMENTED')
      END
```

```
      SUBROUTINE PLOTID(IDF)
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     1       NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2       IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3       VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4       XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5       YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6       AY     ,BY     ,YLPOS  ,
     7       LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8       FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      INTEGER*4 STAR(19)/19*'****'/
      LOGICAL*1 GSPIN, DUM(2)
      INTEGER*4 GSP1,UNITN,OAREA,NULLV(1)
      REAL*8 IDF(9),WORK(36)
      REAL*4 TBLCON(5)/'TABL','E OF',' CON','TENT','S   '/
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6),GLAB(16)
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LOGICAL*1 FA,FB,FC,GMODE1
      EQUIVALENCE  (WORK(11),GLAB(1))
      EQUIVALENCE (RWORK1,Y,X,XINC,YINC)
      GO TO 11111
      WRITE(NOUT,1000)(IDF(I),I=1,9)
C    INITIALIZE GRAPHIC SERVICES
      CALL INGSP(GSP1,NULLV)
C    SPECIFY SUBROUTINE LINK/LOAD STATUS
      CALL SPEC(GSP1,1,30,37,38,40)
      CALL SPEC(GSP1,2,17,-19,53,-56)
C    INITIALIZE THE 2250
      CALL INDEV(GSP1,UNITN,I2250)
C    INITIALIZE GRAPHIC DATA SETS
      CALL INGDS(I2250,IGDSC,OAREA)
      CALL INGDS(I2250,IGDS5,128)
      CALL INGDS(I2250,IGDS6,512)
C    SET NO SCISSIORING FOR ALL DATA SETS
      CALL SSCIS(IGDSC,3)
      CALL SSCIS(IGDS5,3)
      CALL SSCIS(IGDS6,3)
11111 CONTINUE
C    COMPUTE X AND Y TIC MARK COORDINATES AND ADJUST GRID SIZE
C    PARAMETERS FOR TRUNCATION ERRORS
      XINC = (XUR - XLL)/(FLOAT(NXD)-1.0)
      XTIC(1) = XLL
      DO 10 I=2,NXD
   10 XTIC(I) = XTIC(I-1) + XINC
      XUR = XTIC(NXD)
      XSIZE = XUR - XLL
      YSIZE = 0.7*XSIZE
      YLL = YUR - YSIZE
      YINC = YSIZE/(FLOAT(NYD)-1.0)
      YTIC(1) = YLL
      DO 20 I=2,NYD
   20 YTIC(I) = YTIC(I-1) + YINC
      YUR = YTIC(NYD)
      YSIZE = YUR - YLL
C    CREATE TEXT ELEMENTS FOR LIGHT PEN OPTION SELECTION IN IGDS5.
      X = (RU - 21.*HSPB) - 1.0
      CALL STPOS(IGDS5,X,4.5*VSPB)
      CALL PTEXT(IGDS5,GLAB,20,NULLV,IKEY1,2)
      CALL PTEXT(IGDS5,GLAB(6),20,NULLV,IKEY3,2,X,1.5*VSPB)
      CALL PTEXT(IGDS5,GLAB(11),12,NULLV,IKEY2,2,X,3.*VSPB)
      CALL EXEC(IGDS5)
      CALL INCL(IGDS5)
C    CREATE DICTIONARY STATIC ELEMENTS IN IGDS6
      CALL SCHAM(IGDS6,2)
      CALL STPOS(IGDS6,1375.,4095.)
      CALL PTEXT(IGDS6,TBLCON(1),17,-1,NULLV,2)
      CALL PTEXT(IGDS6,GLAB(11),12,-2,NULLV,2,3000.,0.)
      CALL PTEXT(IGDS6,GLAB(14),12,-3,NULLV,2,3000.,1.5*VSPL)
      CALL EXEC(IGDS6)
C    CREATE ATTENTION LEVEL FOR 2250
      CALL CRATL(I2250,IATN)
C    DESIGNATE THAT BOTH CHARACTER CODE AND COORDINATES OF CHARACER
C    DETECTED BY THE LIGHT PEN ARE TO BE RETURNED BY RQATN SUBROUTINE
      CALL MLPEO(IATN,2,4,1)
C    SET L.P. ATTENTIONS AND LIGHT F.K. 0
      CALL SLPAT(IGDS5,1)
      CALL SLPAT(IGDS6,1)
      CALL MLITS(I2250,3)
C    GENERATE ID FRAME IN IGDSC
      CALL STPOS(IGDSC,0.,2290.)
      CALL PTEXT(IGDSC,STAR,74,NULLV,NULLV)
      CALL PTEXT(IGDSC,STAR,74,NULLV,NULLV,1,0.0,1810.)
      CALL PTEXT(IGDSC,IDF,72,NULLV,NULLV,1,56.,2050.)
C    CREATE VERTICAL BOX OUTLINE ELEMENTS
      Y=1810.
      DO 1 I=1,5
      Y=Y+80.
      CALL PTEXT(IGDSC,STAR,1,NULLV,NULLV,1,0.0,Y)
```

```
    1 CALL PTEXT(IGDSC,STAR,1,NULLV,NULLV,1,4088.,Y)                    PLID0089
C     PLACE ID FRAME TITLE IN PLOT DICTIONARY                             PLID0090
      CALL DICTRY(IDF)                                                    PLID0091
 1000 FORMAT('1',T24,'<<< ',9A8,' >>>')                                   PLID0092
      RETURN                                                              PLID0093
      END                                                                 PLID0094
```

```
      SUBROUTINE PLTEND(IDUM)
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     1      NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2      IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3      VSPB   ,VSPL   ,NDUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4      XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5      YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6      AY     ,BY     ,YLPOS  ,
     7      LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8   FC     ,GMODEL,GSPIN  ,DUM    ,IRECAL,IFLAG
      LOGICAL*1 GSPIN, DUM(2)
      REAL*8 END(1)/'END     '/,WORK(36)
      INTEGER*4 GSP1,UNITN,NULLV(1),OAREA
      LOGICAL*1 FA,FB,FC,GMODEL
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      IRECAL = 1
      CALL SHIFT1
      CALL PTEXT(IGDSC,END,4,NULLV,NULLV,1,1879.,2047.)
      CALL EXEC(IGDSC)
      IFLAG=0
      YLPOS=3796.0
      CALL  TMDEV(I2250)
      CALL TMGSP(GSP1)
  100 FORMAT(' ***END OF GRAPHIC PORTION')
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE POINTS(X,Y,NP,NCHAR,IERR,XMIN,XMAX,YMIN,YMAX,PLTID)
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     1      NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2      IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3      VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4      XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5      YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6      AY     ,BY     ,YLPOS  ,
     7      LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8      FC     ,GMODEL,GSPIN  ,DUM    ,IRECAL
      LOGICAL*1 GSPIN, DUM(2)
      REAL*8 ERR3(6)/'OFF SCAL','E POINTS',' ENCOUNT','ERED    ',
     1     2*'        '/,WORK(36),PLTID(2)
      LOGICAL*1 FA,FB,FC,GMODEL
      LOGICAL*1 LOGXSW,LOGYSW,OFSC,NPTF,NRGF,ERRHD
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      REAL*4 XPOS(36),YPOS(36),XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      LOGICAL*1 OFF
      DIMENSION X(NP),Y(NP)
      ISAVE=NP
      OFSC=.FALSE.
      IF(FA)GO TO 102
      IPTNOW=IPTNOW+NP
      IF(IPTNOW-MAXPT)1,1,7
    7 CONTINUE
      NP=NP-(IPTNOW-MAXPT)
      FA=.TRUE.
      WRITE(NOUT,2000)MAXPT
      IF(NP)102,102,1
C     DETERMINE PLOT SYMBOL TO BE USED.  A VALUE OF 0 IS RETURNED
C     IF THE SELECTED PLOT SYMBOL IS 'POINTS' OR A NON-DISPLAYABLE
C     SYMBOL.  IN THE LATTER CASE 'POINTS' IS THE DEFAULT.
    1 CONTINUE
      CALL PLOTSM(NCHAR,ISYM)
      IF(ISYM .EQ. 0)GO TO 5
      ASSIGN 2 TO M
      DO 3 K=1,NP
      OFF=.FALSE.
      IF(LOGXSW)XPOS(1)=AX*ALOG10(X(K)) + BX
      IF(.NOT.LOGXSW)XPOS(1)=AX*X(K) + BX
      IF(LOGYSW)YPOS(1)=AY*ALOG10(Y(K)) + BY
      IF(.NOT.LOGYSW)YPOS(1)=AY*Y(K) + BY
      GO TO 75
    2 CONTINUE
      IF(OFF)GO TO 3
      CALL PTEXT(IGDSC,ISYM,1,NULLV,NULLV,1,XPOS(1),YPOS(1))
    3 CONTINUE
      GO TO 100
    5 CONTINUE
      ASSIGN 80 TO M
      J1 = 1
      N = 1
      NL = NP
      IF(NP-IPT)10,10,20
   10 J2 = NP
      GO TO 70
   20 N = 2
      J2 = IPT
      GO TO 70
   40 NL = NL - IPT
      J1 = J2 + 1
      IF(NL-IPT)60,50,65
   50 N = 1
      GO TO 65
   60 J2 = NP
      N = 1
      GO TO 70
   65 J2 = J2 + IPT
   70 K = J1 - 1
      L = J2 - J1 + 1
      DO 80 I=1,L
      K = K + 1
      IF(LOGXSW)XPOS(I)=AX*ALOG10(X(K)) + BX
      IF(.NOT.LOGXSW)XPOS(I)=AX*X(K) + BX
      IF(LOGYSW)YPOS(I)=AY*ALOG10(Y(K)) + BY
      IF(.NOT.LOGYSW)YPOS(L)=AY*Y(K) + BY
   75 CONTINUE
      IF(X(K) .LT. XMIN .OR. X(K) .GT. XMAX)GO TO 78
      IF(Y(K) .GE. YMIN .AND. Y(K) .LE. YMAX)GO TO 72
   78 CONTINUE
      OFF=.TRUE.
      IF(OFSC)GO TO 79
      OFSC=.TRUE.
      CALL PTEXT(IGDSC,ERR3,48,NULLV,NULLV,1,0.0,YLL-7.0*VSPB)
      IF(ERRHD)CALL PTEXT(IGDSC,PLTID,14,NULLV,NULLV,1,0.,YLL-5.5*VSPB)
      ERRHD=.FALSE.
   79 WRITE(NOUT,1000)X(K),Y(K)
   72 GO TO M,(2,80)
   80 CONTINUE
```

```
   90 CALL PPNT(IGDSC,XPOS,YPOS,NULLV,NULLV,1,L,1,1)                 PNTS0090
      GO TO(100,40),N                                                PNTS0091
  100 CONTINUE                                                       PNTS0092
C    CREATE ZERO POSITION LINES IF ON GRAPH                          PNTS0093
C    X(0)=BX AND Y(0)=BY FOR LINEAR GRAPHS                           PNTS0094
      IF(LOGXSW)GO TO 101                                            PNTS0095
      IF(BX .LE. XLL .OR. BX .GE. XUR)GO TO 101                      PNTS0096
      CALL STPOS(IGDSC,BX,YLL)                                       PNTS0097
      CALL PLINE(IGDSC,BX,YUR)                                       PNTS0098
  101 IF(LOGYSW)GO TO 102                                            PNTS0099
      IF(BY .LE. YLL .OR. BY .GE. YUR)GO TO 102                      PNTS0100
      CALL STPOS(IGDSC,XLL,BY)                                       PNTS0101
      CALL PLINE(IGDSC,XUR,BY)                                       PNTS0102
  102 CONTINUE                                                       PNTS0103
      NP=ISAVE                                                       PNTS0104
      RETURN                                                         PNTS0105
 1000 FORMAT(' ***OFF-SCALE POINT    X = ',E12.5,5X,'Y = ',E12.5)    PNTS0106
 2000 FORMAT(' ***MAX NUMBER POINTS - ',I4,' - EXCEEDED')            PNTS0107
      END                                                            PNTS0108
```

```
C       ROUTINE TO PRINT PL360 GRAPH
C
C        WRITTEN BY C G HOOKS          IBM HUNTSVILLE
C
        SUBROUTINE PR360 (ARRAY)
C
        DIMENSION ARRAY (1540)
        EQUIVALENCE (IEST,A)
        INTEGER *4  TEST /'DAYS'/
        A = ARRAY(1490)
        A1 = ARRAY(1491)
        A2 = ARRAY(1492)
        A3 = ARRAY(1493)
        A4 = ARRAY(1494)
C
C       TEST FOR JUST PRINTING THE FIELD OF ARRAY WITHOUT SCALES
        IJ = 1515
        JI = 1520
        IF ( ARRAY(IJ) - ARRAY(JI) )  41,42,41
     42 CONTINUE
        WRITE (6, 43) ARRAY
     43 FORMAT (28A4)
        RETURN
C
     41 CONTINUE
        WRITE (6,6) (ARRAY(I), I = 1,56)
      6 FORMAT (2A4, 8X, 26A4)
        WRITE (6,9)
      9 FORMAT (16X,1H+, 5(19H-------------------, 1H+) )
        WRITE (6,7) (ARRAY(I), I = 57,1456)
      7 FORMAT(A4,F11.2,1H+,25A4,A1,1H+ / 2A4,7X,1H|,25A4,A1,1H| /
       1             2A4,7X,1H|,25A4,A1,1H| / 2A4,7X,1H|,25A4,A1,1H| /
       2             2A4,7X,1H|,25A4,A1,1H| / 2A4,7X,1H|,25A4,A1,1H| /
       3             2A4,7X,1H|,25A4,A1,1H| / 2A4,7X,1H|,25A4,A1,1H| /
       4             2A4,7X,1H|,25A4,A1,1H| / 2A4,7X,1H|,25A4,A1,1H| )
        WRITE (6,5) (ARRAY(I), I = 1457,1484)
      5 FORMAT(A4,F11.2,1H+,25A4,A1,1H+ )
        IF (TEST .EQ. IEST) GO TO 50
        WRITE (6,8) (ARRAY(I), I = 1515,1520)
      8 FORMAT (16X,      1H+, 5(19H-------------------,1H+) /
       1    9X, F11.0,  5(9X,F11.0) /)
        WRITE (6,6) (ARRAY(I), I = 1485,1512)
        RETURN
     50 CONTINUE
        WRITE(6,11)
        WRITE(6,10)
     11 FORMAT(16X,'+OOOOOOONNNNNNNNNDDD+DDDDJJJJJJJFFFFFF+',
       &          'FMMMMMMMMAAAAAAAAAMM+MMMMMMJJJJJJJJJJJJJ+',
       &          'JJAAAAAAAASSSSSSSSS+')
     10 FORMAT(19X,'OCT.      NOV.      DEC.      JAN.     FEB.    ',
       &          ' MARCH    APRIL    MAY         JUNE      JULY  ',
       &          '  AUG.     SEPT.   ')
        RETURN
        END
```

```
      SUBROUTINE PRECHK(DRGPM,DRHP,DRSF,VWIN,SGRT,NATRH)
C  CHECKS PRECIPITATION-STREAMFLOW ANOMALIES AND ADJUSTS PRECIPITATION
C     WHERE NECESSARY
      DIMENSION  DRGPM(366),DRHP(366,24),DRSF(366)
      INTEGER  DAY,HOUR,SGRT
      AHP = 0.0
      NRHA = 24 - NATRH
      RGPM = DRGPM(90)
      DAY = 90
      RMWR = 1.25
  100 DAY = DAY + 1
      IF(DAY .GT. 200 .OR. VWIN .GT. 750.0) RMWR = 2.00
      RFRISE = (DRSF(DAY) - DRSF(DAY-1))/VWIN
      DO 101 HOUR = 1,24
      IF(HOUR .EQ. SGRT+1) RGPM = DRGPM(DAY)
      AHP = AHP + DRHP(DAY,HOUR)*RGPM
      IF(HOUR .NE. NRHA) GO TO 101
      RWRAIN = AHP
      AHP = 0.0
  101 CONTINUE
      IF(RFRISE .GT. RWRAIN .AND. RFRISE .GT. 0.1) GO TO 102
      IF((RWRAIN .GT. RMWR .AND. RFRISE .LT. 0.02*RWRAIN) .OR. (RWRAIN
     1 .GT. 3.00 .AND. RFRISE .LT. 0.05*RWRAIN)) GO TO 104
      GO TO 108
  102 IF(RWRAIN .GT. 0.05) GO TO 103
      RAA = RFRISE*2.0 - RWRAIN + 1.0
      DRHP(DAY,13) = RAA
      WRITE(6,1) DAY, RAA
    1 FORMAT(/10X,'FOR DAY',I4,1X,'RAIN ADDED OF',F7.2)
      GO TO 108
  103 RAM = 2.0*RFRISE/RWRAIN
      GO TO 105
  104 RAM = 10.0*RFRISE/RWRAIN
  105 IF(RAM .LT. 0.0) GO TO 108
      WRITE(6,2) DAY,RAM,RWRAIN
    2 FORMAT(/5X,'FOR DAY',I4,1X,'RAIN ADJUSTMENT MULTIPLIER IS',F8.4,
     1  1X,'RECORDED RAIN IS',F7.2)
      DO 106 HOUR = 1,NRHA
  106 DRHP(DAY,HOUR) = DRHP(DAY,HOUR)*RAM
      IF(NATRH .EQ. 0) GO TO 108
      NFRHA = NRHA + 1
      DO 107 HOUR = NFRHA,24
  107 DRHP(DAY-1,HOUR) = DRHP(DAY-1,HOUR)*RAM
  108 IF(DAY .NE. 273) GO TO 100
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE PREPRD(RGPM,DRHP,DAY,HOUR,DPY,PRD,PEP,PRH)
C   DIVIDES HOURLY PRECIPITATION TOTALS AMONG PERIODS FOR SMALL BASINS
      DIMENSION DRHP(366,24), PE4P(4)
      INTEGER DAY,DPY,HOUR,PRD
      PEP = 0.0
      IF(PRH .EQ. 0.0) RETURN
      IF(PRD .EQ. 1) GO TO 100
      PEP = PE4P(PRD)
      RETURN
  100 LHOUR = HOUR - 1
      LDAY = DAY
      IF(LHOUR .GE. 1) GO TO 101
      LHOUR = 24
      LDAY = DAY - 1
      IF(LDAY .EQ. 0) LDAY = 365
      IF(LDAY .EQ. 365) LDAY = 59
      IF(LDAY .EQ. 59 .AND. DPY .EQ. 366) LDAY = 366
  101 PRLH = RGPM*DRHP(LDAY,LHOUR)
      NHOUR = HOUR + 1
      NDAY = DAY
      IF(NHOUR .LE. 24) GO TO 102
      NHOUR = 1
      CALL DAYNXT(NDAY,DPY)
  102 PRNH = RGPM*DRHP(NDAY,NHOUR)
      IF(PRH .GT. PRLH .AND. PRH .GT. PRNH) GO TO 103
      GO TO 104
  103 PE4P(1) = 0.10
      PE4P(2) = 0.28
      PE4P(3) = 0.46
      PE4P(4) = 0.16
      GO TO 108
  104 IF(PRH .LT. PRLH .AND. PRH .LT. PRNH) GO TO 105
      GO TO 106
  105 PE4P(1) = 0.28
      PE4P(2) = 0.10
      PE4P(3) = 0.16
      PE4P(4) = 0.46
      GO TO 108
  106 IF(PRNH .GE. PRLH) GO TO 107
      PE4P(1) = 0.46
      PE4P(2) = 0.16
      PE4P(3) = 0.28
      PE4P(4) = 0.10
      GO TO 108
  107 PE4P(1) = 0.10
      PE4P(2) = 0.28
      PE4P(3) = 0.16
      PE4P(4) = 0.46
  108 DO 109 KPRD = 1,4
  109 PE4P(KPRD) = PE4P(KPRD)*PRH
      PEP = PE4P(1)
      RETURN
      END
```

```
      SUBROUTINE RTVARY(CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCTRI,EXQPV,
     1   LSHFT,TFCFS)
      DIMENSION  AWSBIT(99),BTRI(99),CTRI(99),SATRI(99)
      LOGICAL  LSHFT
      DO 100 KIA = 1,MXTRI
      SATRI(KIA) = 0.0
  100 AWSBIT(KIA) = 0.0
      LSHFT = .FALSE.
      FMXTRI = MXTRI
      FNBTRI = NBTRI
      FNPTRI = NCTRI
      TFX = TFCFS
      TFMRT = 0.1*CHCAP
      IF(TFX .LT. TFMRT) TFX = TFMRT
      IF(FNPTRI .EQ. FMXTRI .AND. TFX .EQ. TFMRT) RETURN
      FNTRI = FNBTRI*(CHCAP/TFX)**EXQPV + 0.5
      IF(FNTRI .LT. 1.0) FNTRI = 1.01
      NCTRI = FNTRI
      FNSTRI = NCTRI
      IF(FNSTRI .NE. FNPTRI) LSHFT = .TRUE.
      IF(.NOT. LSHFT) RETURN
      IF(FNPTRI .GT. 98.5) GO TO 101
      FCNTRI = ABS(FNSTRI - FNPTRI)
      IF(FCNTRI .LE. 1.1) GO TO 101
      IF(FNSTRI .GT. FNPTRI) FNSTRI = FNPTRI + 1.0
      IF(FNSTRI .LT. FNPTRI) FNSTRI = FNPTRI - 1.0
      NCTRI = FNSTRI
  101 KB1 = 0
      KB2 = 1
      KB3 = 0
  102 KB1 = KB1 + 1
      IF(KB1 .GT. NBTRI) GO TO 105
      KB4 = 0
      WSBIT = BTRI(KB1)/FNSTRI
  103 KB4 = KB4 + 1
      IF(KB4 .GT. NCTRI) GO TO 102
      AWSBIT(KB2) = AWSBIT(KB2) + WSBIT
      KB3 = KB3 + 1
      IF(KB3 .LT. NBTRI) GO TO 104
      KB3 = 0
      KB2 = KB2 + 1
  104 GO TO 103
  105 IF(FNPTRI .GT. 98.5) GO TO 108
      DO 107 KB6 = 1,NCTRI
      DO 106 KB7 = 1,KB6
  106 SATRI(KB6) = SATRI(KB6) + AWSBIT(KB7) - CTRI(KB7)
  107 CONTINUE
  108 DO 109 KB5 = 1,MXTRI
  109 CTRI(KB5) = AWSBIT(KB5)
      RETURN
      END
```


GE IS
LITY

```
      SUBROUTINE SGRID
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     1       NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2       IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3       VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4       XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5       YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6       AY     ,BY     ,YLPOS  ,
     7       LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8       FC     ,GMODE1,GSPIN  ,DUM    ,IRFCAL
      LOGICAL*1 GSPIN, DUM(2)
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      REAL*8 WORK(36)
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LOGICAL*1 FA,FB,FC,GMODE1
      EQUIVALENCE (RWORK1,DEL)
C    CREATE GRID BOX ELEMENTS
      CALL STPOS(IGDSC,XLL,YLL)
      CALL PLINE(IGDSC,XUR,YLL)
      CALL PLINE(IGDSC,XUR,YUR)
      CALL PLINE(IGDSC,XLL,YUR)
      CALL PLINE(IGDSC,XLL,YLL)
      DEL = .5*VSPB
C    CREATE LOWER TIC MARK ELEMENTS
      CALL PSGMT(IGDSC,XTIC,YLL-DEL,XTIC,YLL+DEL,NULLV,NULLV,1,NXD,
     1         1,0,1,0)
C    CREATE UPPER TIC MARK ELEMENTS
      CALL PSGMT(IGDSC,XTIC,YUR-DEL,XTIC,YUR+DEL,NULLV,NULLV,1,NXD,
     1         1,0,1,0)
C    CREATE LEFT TIC MARK ELEMENTS
      CALL PSGMT(IGDSC,XLL-DEL,YTIC,XLL+DEL,YTIC,NULLV,NULLV,1,NYD,
     1         0,1,0,1)
C    CREATE RIGHT TIC MARK ELEMENTS
      CALL PSGMT(IGDSC,XUR-DEL,YTIC,XUR+DEL,YTIC,NULLV,NULLV,1,NYD,
     1         0,1,0,1)
      RETURN
      END
```

```
      SUBROUTINE SHIFT1
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     1       NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2       IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3       VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4       XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5       YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6       AY     ,BY     ,YLPOS  ,
     7       LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8       FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      LOGICAL*1 GSPIN, DUM(2)
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      LOGICAL*1 FA,FB,FC,GMODE1
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      REAL*8 WORK(36)
      IABSDP=IABS(IDSP)+1
      IF (IRECAL .EQ. 0) CALL OMIT(IGDSC)
      CALL EXEC(IGDSC)
      CALL BUFOUT(I2250,IABSDP,IRC)
      IF (IRC .NE. 0) WRITE(NOUT,1040) IRC
      IF (IRECAL .EQ. 0) GO TO 17
      IF(IDSP)11,12,11
   11 CALL INCL(IGDS5,NULLV,IKEY3)
   12 CALL ATTEN
      CALL OMIT(IGDS5,NULLV,IKEY3)
   17 CALL RESET(IGDSC)
      IDSP=IABSDP
   14 RETURN
 1040 FORMAT(' BUFFER READ ERROR - R.C. = ',Z8)
      END
```

```
      SUBROUTINE SNOMEL(BODESM,SPTWCC,SPM,ELDIF,DAY,SPBFLW,XDNFS,FFOR,
     1  FFSI,MRNSM,DSMGH,SDEPTH,STMP,PXCSA,HOUR,SAX,SOFRF,OFRFIS,SOFRFI,
     2  AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,WT4AM,WT4PM,ASM,ASMRG,
     3  SASFX,SARAX,DMXT,DMNT,RICY,FIRR)
C  SNOWMELT COMPUTATION
      DIMENSION  DMNT(366),DMXT(366),FIRR(15),RICY(37)
      INTEGER  DAY,HOUR
      REAL  MHSM,MRNSM
      IF((DAY .NE. 274) .OR. (HOUR .NE. 1)) GO TO 100
      SPLW = 0.0
      XELR = 0.0
      SDSC = 0.0278
      FDSC = 0.0
      FTA = 0.0
      RICD = 0.0
      KRIA = 0
  100 CONTINUE
C  CALCULATION OF HOURLY AIR TEMPERATURE
C    DMXT CURRENT DAY, DMNT NEXT DAY
      IF(HOUR .NE. 4) GO TO 101
      FDSC = 0.0
      FTA = FDSC
      WT4PM = DMXT(DAY) - 4.0*ELDIF + (XELR/4.0)*0.7*ELDIF
  101 IF(HOUR .EQ. 10) SDSC = -0.0278
      IF(HOUR .EQ. 22) SDSC = 0.0278
      IF(HOUR .NE. 16) GO TO 102
      NDAY = DAY + 1
      IF(NDAY .EQ. 366) NDAY = 1
      IF(NDAY .EQ. 60 .AND. DMXT(366) .NE. 0.0) NDAY = 366
      IF(NDAY .EQ. 367) NDAY = 60
      WT4AM = DMNT(NDAY) - (XELR/4.0)*3.3*ELDIF
  102 IF(PRH .LE. 0.0 .OR. XELR .GE. 4.0) GO TO 103
      WT4AM = WT4AM - 0.825*ELDIF
      WT4PM = WT4PM + 0.175*ELDIF
      XELR = XELR + 1.0
  103 IF(PRH .NE. 0.0 .OR. XELR .LE. 0.0) GO TO 104
      WT4AM = WT4AM + 0.825*ELDIF
      WT4PM = WT4PM - 0.175*ELDIF
      XELR = XELR - 1.0
  104 TEH = WT4AM + FTA*(WT4PM - WT4AM)
      FDSC = FDSC + SDSC
      FTA = FTA + FDSC
      IF(PRH+SPTW .EQ. 0.0) GO TO 128
      IF(HOUR .NE. 24) GO TO 105
C  CALCULATION OF TIME AGING OF THE SNOWPACK
      SAX = SAX + 1.0
      IF(SAX .GT. 15.0) SAX = 15.0
  105 IF(TEH .GT. 32.0) GO TO 110
C  PRECIPITATION IN FORM OF SNOW - CALCULATE INTERCEPTION DENSITY OF NEW
C    SNOW COMPACTION, AND SETTLING SNOW PACK AND THE EFFECT ON ALBEDO
      IF(PRH .LE. 0.0) GO TO 110
      PRH = SPM*PRH
      HSF = PRH
      ASM = ASM + HSF
      PRH = (1.0 - (FFSI*FFOR))*PRH
      HSFRG = PRH
      ASMRG = ASMRG + HSFRG
      FSIL = FFSI*FFOR*HSF
      AMFSIL = AMFSIL + FSIL
      IF(TEH .LE. 0.0) GO TO 106
      DNFS = XDNFS + ((0.01*TEH)**2)
      GO TO 107
  106 DNFS = XDNFS
  107 IF(SPTW .GT. 0.0 .AND. SDEPTH .GT. SPTW) SDEPTH = SDEPTH - ((PRH*
     1  SDEPTH/SPTW)*((0.10*SDEPTH)**0.25))
      SPTW = SPTW + PRH
      SDEPTH = SDEPTH + (PRH/DNFS)
      SASFX = SASFX + PRH
      IF(SASFX .GE. PXCSA) GO TO 108
      GO TO 109
  108 SAX = SAX - 1.0
      IF(SAX .LT. 0.0) SAX = 0.0
      SASFX = SASFX - PXCSA
  109 PRH = 0.0
  110 CONTINUE
      IF(SPTW .LE. 0.0) GO TO 127
C  SEASONAL MELT FACTOR ADJUSTMENT
      FDAY = DAY
      KAAO = KRIA
      KRIA = 1.0 + (FDAY/10.0)
      IF(KAAO .NE. KRIA) RICD = RICY(KRIA)
      IF(TEH .LE. 32.0) GO TO 111
      GO TO 114
C  CALCULATION OF NEGATIVE MELT
  111 IF(TANSM .LE. 11.5*MRNSM) GO TO 112
      IF(TANSM .LT. 1.0) TANSM = TANSM + ((5.0*MRNSM)**(1.3 + 2.0*
     1  TANSM))
      GO TO 113
  112 TANSM = TANSM + MRNSM
```

```
  113 IF(TANSM .GT. 0.08*SPTW) TANSM = 0.08*SPTW
      GO TO 127
C     EFFECT OF RAIN ON ALBEDO
  114 SARAX = SARAX + PRH
      IF(SARAX .LT. PXCSA/2.0) GO TO 115
      SAX = SAX + 1.0
      IF(SAX .GT. 15.0) SAX = 15.0
      SASFX = 0.0
      SARAX = SARAX - (PXCSA/2.0)
  115 IF(TEH .GT. 32.0) HSM = (TEH -32.0)*BDDFSM
      IF(TEH .LT. 32.0) HSM = 0.0
      HSM = HSM*RICD
      KAA = 1.0 + SAX
      IF(SAX .LT. 15.0) HSM = HSM*(1.0 - ((1.0 - FFOR)*FIKR(KAA)))
      IF(SAX .EQ. 15.0) HSM = HSM*(1.0 - ((1.0 - FFOR)*FIKR(15)))
      IF(PRH .GT. 0.0) HSM = HSM + ((TEH - 32.0)*(PRH/144.0))
      IF(STMD .GT. 0.3 .AND. SPTW .LT. SPTWCC) GO TO 116
      GO TO 117
  116 MHSM = HSM
      HSM = (SPTW/SPTWCC)*HSM
      IF(HSM .LT. 0.1*MHSM) HSM = 0.1*MHSM
  117 IF(HSM .LT. SPTW) GO TO 118
      HSM = SPTW
      SDEPTH = 0.0
      SPTW = 0.0
      SPLW = 0.0
      RICD = 0.0
      TANSM = 0.0
      SAX = 15.0
      OFRF = SOFRF
      OFRFIS = SOFRFI
      GO TO 122
  118 SPTW = SPTW - HSM
      IF(SFMD .LE. 0.0) GO TO 122
      IF(SAX .GE. 15.0) GO TO 121
      IF(SAX .GE. 6.0) GO TO 119
      SDEPTH = SDEPTH - (HSM/(0.5*SFMD))
      GO TO 122
  119 IF(SAX .LE. 10.0) GO TO 120
      SDEPTH = SDEPTH - (HSM/(0.9*SFMD))
      GO TO 122
  120 SDEPTH = SDEPTH - (HSM/(0.7*SFMD))
      GO TO 122
  121 SDEPTH = SDEPTH - (HSM/SFMD)
  122 CONTINUE
      IF(SPTW .LT. 0.00001) SPTW = 0.0
C     CALCULATION OF LIQUID-WATER-HOLDING CAPACITY
      SPLWC = SPRFLW*SPTW
      IF(SFMD .GT. 0.6) SPLWC = SPRFLW*(3.0 - 3.33*SFMD)*SPTW
      IF(SPLWC .LT. 0.0) SPLWC = 0.0
C     ACCOUNTING OF MELT WATER AND RAIN
      IF((SPLW + HSM + PRH) .GT. (SPLWC + TANSM)) GO TO 123
      GO TO 124
  123 PRH = HSM + PRH + SPLW - SPLWC - TANSM
      SPLW = SPLWC
      SPTW = SPTW + TANSM
      TANSM = 0.0
      GO TO 127
  124 IF((HSM + PRH) .LE. TANSM) GO TO 126
  125 SPTW = SPTW + TANSM
      SPLW = SPLW + HSM + PRH - TANSM
      PRH = 0.0
      TANSM = 0.0
      GO TO 127
  126 TANSM = TANSM - HSM - PRH
      SPTW = SPTW + HSM + PRH
      PRH = 0.0
  127 CONTINUE
      HSM = 0.0
C     CALCULATION OF DENSITY AND ADJUSTMENT OF OVERLAND FLOW TIME
      IF(SDEPTH .LE. 0.0 .OR. SPTW .GE. SDEPTH) GO TO 128
      STMD = (SPTW + SPLW)/SDEPTH
      SFMD = SPTW/SDEPTH
      OFRF = 0.33*SOFRF
      IF(SPTW .LE. SPTWCC) OFRF = (1.0 - (SPTW/SPTWCC)*0.67)*SOFRF
  128 IF(SDEPTH .LE. 0.0) OFRF = SOFRF
      OFRFIS = SOFRFI*OFRF/SOFRF
C     CALCULATION OF GROUNDMELT
      IF( HOUR .NE. 12 .OR. SPTW .LE. 0.0) RETURN
      IF(SPTW .LE. DSMGH) GO TO 129
      PRH = PRH + DSMGH
      SPTW = SPTW - DSMGH
      IF(STMD .LT. 0.50 .AND. SDEPTH .GT. 2.0*DSMGH) SDEPTH = SDEPTH -
     1  2.0*DSMGH
      RETURN
  129 PRH = SPTW + PRH + SPLW
      TANSM = 0.0
      RICD = 0.0
      SPLW = 0.0
```

```
      SDEPTH = 0.0                                                   SNOW0179
      SPTW = 0.C                                                     SNOW0180
      SAX = 15.0                                                     SNOW0181
      OFRF = SOFRF                                                   SNOW0182
      OFRFIS = SOFRFI                                                SNOW0183
      RETURN                                                         SNOW0184
      END                                                            SNOW0185
```

```
      SUBROUTINE  STAT(R,S,N,AREA,MAXR,MAXS,SSQ,SSQI,SUMD,
     1                 SUMSQ,MEANR,MEANS,VARR,VARS,CORCF)
C
C
      INTEGER   I,N
      REAL   R(N),S(N),AREA,SUMS,SUMR,MAXS,MAXR,SSQ,SSQI,SUMD,SUMSQ
      REAL   MEANS, MEANR, VARS, VARR, CORCF
      REAL   TEMP, TEMPS, TEMPR, AS, AR, ASR
C
C
C    INPUT:  S = ARRAY OF SIMULATED READINGS
C            R = ARRAY OF OBSERVED READINGS
C            N = NUMBER OF READINGS
C            AREA = AREA OF WATERSHED
C
C    OUTPUT: SUMS = SUM OF SIMULATED READINGS
C            SUMR = SUM OF OBSERVED READINGS
C            MAXS = MAXIMUM OF SIMULATED READINGS
C            MAXR = MAXIMUM OF OBSERVED READINGS
C            SSQ = SUM SQUARED WITH WEIGHT OF 20
C            SSQI = SUM SQUARED WITH VARIABLE WEIGHT
C            SUMD = SUM OF (OBSERVED - SIMULATED) READINGS
C            SUMSQ = SQRT OF SUM OF (OBSERVED - SIMULATED)**2)
C            MEANS = MEAN OF SIMULATED READINGS
C            MEANR = MEAN OF OBSERVED READINGS
C            VARS = VARIANCE OF SIMULATED READINGS
C            VARR = VARIANCE OF ACTUAL READINGS
C            CORCF = CORRELATION COEFFICIENT
C
C
      SUMS  = 0.
      SUMR  = 0.
      MAXS = S(1)
      MAXR  = R(1)
      SSQ  = 0.
      SSQI  = 0.
      SUMD  = 0.
      SUMSQ  = 0.
C
      DO  100  I= 1, N
      TEMPS  = S(I)
      TEMPR  = R(I)
C
C     COMPUTE SUM OF READINGS
C
      SUMS  =  SUMS + TEMPS
      SUMR  =  SUMR + TEMPR
C
C     FIND MAX OF READINGS
C
      MAXS  =  AMAX1(MAXS,TEMPS)
      MAXR  =  AMAX1(MAXR,TEMPR)
C
C     COMPUTE SUM SQUARES
C
      IF  (I .EQ. 1)  GO TO 50
      WEIGHT  =  20.
      IF  (N  .GT.  12.)  WEIGHT  =  WEIGHT / 30.
      TEMP  =  FLOWDI(TEMPS,TEMPR,WEIGHT)
      SSQ  =  SSQ + TEMP * TEMP
      WEIGHT  =  AREA  +  10.
      IF  (N  .GT.  12.)  WEIGHT  =  WEIGHT / 30.
      TEMP  =  FLOWDI(TEMPS,TEMPR,WEIGHT)
      SSQI  =  SSQI + TEMP * TEMP
C
C     COMPUTE SUM OF DELTAS & SUM OF DELTAS SQUARED
C
50    TEMP=ABS(TEMPR)-ABS(TEMPS)
      SUMD  =  SUMD + TEMP
  100 SUMSQ  =  SUMSQ + TEMP * TEMP
C
C
C     COMPUTE MEANS
C
      MEANS  =  SUMS / N
      MEANR  =  SUMR / N
C
C     COMPUTE SQRT(SUM(DELTA**2)
C
      SUMSQ  =  SQRT(SUMSQ)
C
C
      AS  =  0.
      AR  =  0.
      ASR  =  0.
      VARS  =  0.
      VARR  =  0.
C
C
```

```
      DO  200  I= 1, N
C
C     COMPUTE INTERMEDIATE VARIANCE & CORRELATION COEFFICIENT
C
      TEMPS  =   S(I) - MEANS
      TEMP   =   TEMPS * TEMPS
      AS   =   AS + TEMP
      VARS   =   VARS + TEMP
C
      TEMPR  =   R(I) - MEANR
      TEMP   =   TEMPR * TEMPR
      AR   =   AR + TEMP
      ASR   =   ASR + TEMPS * TEMPR
  200 VARR   =   VARR + TEMP
C
C     COMPUTE VARIANCE & CORRELATION COEFFICIENT
C
      VARS   =   VARS / (N - 1)
      VARR   =   VARR / (N - 1)
      CORCF  =   ASR / SQRT(AS*AR)
      RETURN
      END
```

APPENDIX C

PROGRAM FLOW CHARTS

APPENDIX C

PROGRAM FLOW CHARTS

```
MAIN ********  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 197
      C        BASED ON STANFORD WATERSHED WATERSHED MODELS III & IV
      C
      C     ISFLAG IS SET TO BRANCH AS FOLLOWS:
      C        1 - BRANCH TO NOMINAL KWM INITIAL RUN
      C        2 - PLOTS ONLY, NO NEXT YEAR OF DATA
      C        3 - NO PLOTS, NEXT YEAR DATA WITH NEW WATERSHED PARAMETERS
      C        4 - NO PLOTS, NEXT YEAR DATA WITH OUT NEW WATER SHED PARAMETERS
      C        5 - PLOTS , NEXT YEAR DATA WITH NEW WATERSHED PARAMETERS
      C        6 - PLOTS, NEXT YEAR DATA WITHOUT NEW WATERSHED PARAMETERS
      C       7 - PLOTS, WITH SIM AND OBS SUPERIMPOSED ONLY(BOTH MONTH AND YER
      C       8 - PLOTS, WITH SIM AND OBS SUPERIMPOSED ONLY(MONTHS SCALE)
      C       9 - NO PLOTS, NO NEXT YEAR DATA
      C
      C

            DIMENSION  TITLE(18), DRSF(366), DSSF(366), CONOPT(16),
             THSFD(744,3),PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6),Q(337),
             TMRTF(12),TMSTF(15,3),
              CTRI(99),FIRR(15),RICY(37),DPSE(366),BTRI(99)

            COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
            KFLAG,IDFLAG,
            IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
              NCTRI,CTRI,FIRR,RICY,DPSE,BDDFSM,SPBFLW,SPTWCC,SPM,FLDTF,
              XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
             SATRI(99),UHFA(99),
            MNRD,
              FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIP,
              BIVF,OFSS,OFSL OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,
              BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
              IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAET,FPER,
              TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPDR,OFUS,OFUSIS,OFR,OFRIS,PETS,
              RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
              WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQUF,EQDFIS,SOFRF,SOFRFI,
              SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTN,STMD,SFMD,ASMPG,
            DEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
            TCMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
            TSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
            TCRARY(1,6,1),
            DRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)

            COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
            TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSDP,DPET,EDLZS,
            EPCM,SERA,SERR,SESF,SOER,THSF,TMFSIL,TMNET,THOF,TMPET,TMRPM,TMSNE,
            TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,
            TFMAXY,UZC,AEIX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
            TRHF,
            SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
```

```
      DIMENSION RPLOTC(1832),RCOMMA(12087)

      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))

      DIMENSION CRFMI(22),DDIW(366),
        DMNT(366), DMXT(366),DRGPM(366), DRHP(366,24),
        DRSGP(366), DPET(366),EDLZS(366),
       EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
       EMUZC(15,3),EMUZS(15,3),EPCM(12),
       SERA(22), SERR(22), SESF(22), SQER(22),
       THSF(24),TMSF(15,3),TMFSIL(12),TMIF(15,3),TMNFT(12),
       TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMSE(15,3),
       TMSNF(12),TMSTFI(15,3),T200FH(21),T20PRH(21),
       TMRTFI(12),JULDAT(6),TFMAXY(366)

      REAL LZS,IFS

      REAL MNRD

      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY

      INTEGER PEAKS,PHRS,CONOPT,NSPTS

      INTEGER SINDEX,SINDET,MPDAY(15)

      INTEGER NUNIT/2/,IER

      INTEGER ICCN/1/

      REAL SUM(3),APREC(3),QOUT(15)

          INTEGER MONTHS(64),PMONTH,PDATE,EMONTH,EDATE,WYEAR,DATES,PDAY

      DATA MONTHS/'JAN ',0,0,0,4,'FEB ',31,31,31,5,'MAR ',59,59,59,6,
      'APR ',90,90,90,7,'MAY ',120,120,120,8,'JUN ',151,151,151,9,
      'JUL ',181,181,181,10,'AUG ',212,212,212,11,'SEPT',243,243,243,12,
      'OCT ',273,273,274,1,'NOV ',304,304,305,2,'DEC ',334,334,335,3,
      'JAN ',365,365,366/
                  |
        +------------------------------------------------+
       + READ (5,9)                                     +
      +   TITLE                                        +
     +                                                +
    +------------------------------------------------+
                  |
                                          FORMAT     9 FORMAT(18A4)

      CALL   DSUPDT
                  |
             ---------
             |806    |
             ---------

805   CONTINUE
      BACKSPACE 5
      BACKSPACE 5
      REWIND 11
      REWIND 18
      CALL TAPCON(NUNIT,ICON,IER)
      CALL DSUPDT
806   CONTINUE
                  |
      ----------------------------------------
      | ISFLAG =1                            |
      ----------------------------------------
                  |
      ----------------------------------------
      | JPLOT=0                              |
      | IOFLAG =0                            |
      | IENDFG =0                            |
      | ID     =1                            |
      | ID     =5                            |
      ----------------------------------------
                  |
```

```
WRITE (6,9999)
  IDFLAG,IENDFG,ISFLAG,ID

CALL READ(IRSTRT)

CALL READ(PMONTH,PDATE,EMONTH,EDATE,NYEAR,WYEAR)

DO LOOP TO
STMT #    101
KRD=1,16

101 CALL READ(CONOPT(KRD))

PMONTH =((PMONTH-1)*5)+1
MSBDIC =MONTHS(PMONTH)
LEAPY=PMONTH+1

IF
MOD(WYEA
R,4).EQ.0           T    LEAPY=PMONTH+2
F

IF
MOD(NYEA
R,4).EQ.0           T    LEAPY=PMONTH+3
F

DATES=MONTHS(LEAPY)+PDATE
PDAY =(MONTHS(LEAPY))
MONTH=MONTHS(PMONTH+4)
EMONTH =((EMONTH-1)*5)+1
LDAY =MONTHS(LEAPY+5)

IF
PMONTH.E
Q.EMONTH            T    EDATE=MONTHS(LEAPY)+EDA
                         TE
F

(PMONTHIF((PMONTH.NE.EMONTH).AND.(MONTHS(EMONTH+4).NE.4))
        EDATE=(MONTHS(LEAPY+5)+EDATE)

(PMONTHIF((PMONTH.NE.EMONTH).AND.(MONTHS(EMONTH+4).EQ.4))
        EDATE=MONTHS(2)+EDATE

IF
(MONTH.E
Q.5).AND.(EDATE.EQ.
60)                 T    EDATE=366
F

ISTART =CONOPT(15)
NFP  =3
```

```
                 |
           . * * * .
         *    IF     * .  T  +----------------------------+
       *  CONOPT(1       *++++|        NFP=1               |
      *5).EQ.2           *    |                            |
         *  .       . *       +----------------------------+
            *   . *                                 +
            F   |++++++++++++++++++++++++++++++++++++
                |
CALL READ(DT,AREAI,NQ,LOCKUP,IPLOT)
                |
--------------------------------------------------
| ID    =6                                        |
--------------------------------------------------
                |
          +-----------------+
          + DO LOOP TO      +
++++++++++++ STMT #    503  +
+         + I=1,NQ          +
+         +-----------------+
+               |
+ CALL READ (Q(I))
+
503 ++++++++CONTINUE
                |
        +--------------------------------------------+
       + WRITE(6,9990)                              +
      +        IDFLAG,IENDFG,ISFLAG,ID             +
     +                                            +
    +--------------------------------------------+
                |
                          FORMAT 9990 FORMAT(/3X,'     IDFLAG=    ',I5,
'     IENDFG=   ',I5,'     ISFLAG=   ',I5,'    MAIN=    ',I5)
20      GO TO (1,2,3),ISTART
                |
--------------------------------------------------
| ISFLAG =0                                       |
| ID     =2                                       |
--------------------------------------------------
                |
        +--------------------------------------------+
       + WRITE(6,9990)                              +
      +        IDFLAG,IENDFG,ISFLAG,ID             +
     +                                            +
    +--------------------------------------------+
                |
CALL KWMAIN(ISFLAG)
CALL WORSTC(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
CALL  ZEROP(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
CALL FORCST(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)
CALL INT(NQ,Q,DT,AREAI,LOOKUP,IPLOT,NFP,MSBDIC,MPDAY,DIFFS,
DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,DIFFPR,DIFPP,QOUT,
QMAX,PHRO)
CALL OUTPUT(DIFFS,DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,
DIFFPR,DIFFPP,QOUT,QMAX,PHRO,MSBDIC,MPDAY)
                |
           . * * * .
         *    IF     * .  T  +----------------------------+
       *  IRSTRT.E       *++++|    GO TO 805               |
      *Q.1               *    |                            |
         *  .       . *       +----------------------------+
            *   . *
            F   |
STOP
16      CONTINUE
1       CONTINUE
```

```
        ID    =6

        WRITE(6,9990)
              IDFLAG,IENDFG,ISFLAG,ID

        CALL  WORSTC(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
        CALL   ZEROP(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)
        CALL  FORCST(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)
        CALL  INT(NQ,Q,DT,AREAI,LOOKUP,IPLOT,NFP,MSBDIC,MPDAY,DIFFS,
        DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,DIFFPR,DIFPP,QOUT,
        QMAX,PHRO)
        CALL  OUTPUT(DIFFS,DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,
        DIFFPR,DIFFPP,QOUT,QMAX,PHRO,MSBDIC,MPDAY)

             IF           T
         IRSTRT.EQ.1   *+++++   GO TO 805
              F

        STOP
17      CONTINUE

         20

2       CONTINUE
        CALL  PASTRN(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)
        CALL  INT(NQ,Q,DT,AREAI,LOOKUP,IPLOT,NFP,MSBDIC,MPDAY,DIFFS,
        DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,DIFFPR,DIFPP,QOUT,
        QMAX,PHRO)
        CALL  OUTPUT(DIFFS,DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,APREC,
        DIFFPR,DIFFPP,QOUT,QMAX,PHRO,MSBDIC,MPDAY)

             IF           T
         IRSTRT.EQ.1   *+++++   GO TO 805
              F

        STOP
3       CONTINUE

        READ (11)
         (RPLOTC(I),I=1,1832)

        READ (18)
         (RCOMMA(I),I=1,12087)
```

```
                    |
CALL READ (GWS,UZS,LZS,BFNX,IFS,UZC,SIAM)
REWIND 11
REWIND 18
                    |
     +-------------------------------------------------------+
    +  WRITE (11)                                           +
 --+--  (RPLOTC(I),I=1,1832)                               +
  +                                                       +
 +-------------------------------------------------------+
                    |
     +-------------------------------------------------------+
    +  WRITE (18)                                           +
   +    (RCOMMA(I),I=1,12087)                              +
  +                                                       +
 +-------------------------------------------------------+
                    |
                 -------
                 |1    |
                 -------

END
```

```
SUBROUTINE KWMAIN(IBFLAG)

      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
     KFLAG,IDFLAG,
     IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
       NCTRI,CTRI,FIRR,RICY,DPSE,BOUFSM,SPBFLW,SPTWCC,SPM,ELDIF,
       XONFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
      SATRI,UHFA,
     MNRD,
       FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWFTF,SIAC,BMIR,
       BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,
       BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
       IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAET,FPER,
       TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPDR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
       RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
       WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
       SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
     DEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
     TOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
     TSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRAKY(1,6,1),
     TORARY(1,6,1),
     DRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)

      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
     TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
     EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
     TMSTFI,T20QFH,T20PRH,TMRTFI,JULDAT,
     TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
     TRHF,
     SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS

C  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)

C     BASED ON STANFORD WATERSHED MODELS III & IV

      DIMENSION BTRI(99),CONOPT(16),CRFMI(22),CTRI(99),DDIW(366),
       DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
       DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
      EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
      EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
       RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),
      THSF(24),TITLE(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
      TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
      TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T20QFH(21),T20PRH(21),
      UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
      PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)

      DIMENSION RPLOTC(1832),RCOMMA(12087)

      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))

      LOGICAL  LSHFT

      INTEGER   CDSDR,CN,CONOPT,DATE,DAY,DPY,EHSGD,HOUR,HRF,HRL,PDAY,
       PRD,RHPD,RHPH,RSBC,SGMC,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX

      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY

      REAL  IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT

      DATA  MEDCY/  0,        31,59,90,120,151,181,212,243,273,304,334/

      DATA  MEDWY/304,334,365,31,59,90,120,151,181,212,243,273        /

      REAL      MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD

      REAL  SSQM,SSQDI,SSQMI,VDRSF,VDSSF

      REAL  VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
           SMSQM

      REAL  MDRSF,MDSSF,MMRSF,MMSSF

98              IF
             (IBFLAG)

     -             0            +
  -------      -------      -------
  |99   |      |100  |      |116  |
  -------      -------      -------
```

```
                        |
           -------------------------------------------
 99        | IBFLAG =13                              |
           -------------------------------------------
                        |
           RETURN
100        CCNTINUE
           CALL READ(GWS,UZS,LZS,BFNX,IFS,UZC,SIAM)
                        |
               +------------+
               + DO LOOP TO +
     +++++++++++ STMT #  102 +
     +         + KIA = 1,99 +
     +         +------------+
     +                  |
     +     -------------------------------------------
     +     | SATRI(KIA) =0.0                         |
     +     | CTRI(KIA) =0.0                          |
     +     | BTRI(KIA) =0.0                          |
     +     -------------------------------------------
     +                  |
     +     -------------------------------------------
102  ++++++++| UHFA(KIA) =0.0                        |
           -------------------------------------------
                        |
           CALL READ(NCTRI)
                        |
               +------------+
               + DO LOOP TO +
     +++++++++++ STMT #  103 +
     +         + KRD = 1, NCTRI
     +         +------------+
     +                  |
103  ++++++++CALL READ(CTRI(KRD))
                        |
                  . * * .                  -----------------------
              . *    IF     * .   T        |   GO TO 110         |
            *   CONOPT(7      *+++++       |                     |
            *) .NE. 1         *            |                     |
              *           . *              -----------------------
                  * . *
                  F |
                    |
               +------------+
               + DO LOOP TO +
     +++++++++++ STMT #  104 +
     +         + KRD = 1,15 +
     +         +------------+
     +                  |
104  ++++++++CALL READ(FIRR(KRD))
                        |
               +------------+
               + DO LOOP TO +
     +++++++++++ STMT #  105 +
     +         + KRD = 1, 37 +
     +         +------------+
     +                  |
105  ++++++++CALL READ(RICY(KRD))
                        |
               +------------+
               + DO LOOP TO +
     +++++++++++ STMT #  106 +
     +         + KRD = 274,360,10
     +         +------------+
     +                  |
106  ++++++++CALL READ(DPSE(KRD))
                        |
               +------------+
               + DO LOOP TO +
     +++++++++++ STMT #  107 +
     +         + KRD = 1,273,10
     +         +------------+
     +                  |
107  ++++++++CALL READ(DPSE(KRD))
                        |
```

```
                +-------------+
                +  DO LOOP TO  +
     +++++++++++ STMT #   109 +
     +          + IDAY2 = 1, 9 +
     +          +-------------+
     +                |
     +          +-------------+
     +          +  DO LOOP TO  +
     +++++++++++ STMT #   108 +
     +          + IDAY1 = 274,360,10
     +          +-------------+
     +                |
     +       -----------------------------------------
     +       | DAY   =IDAY1+IDAY2                    |
     +       -----------------------------------------
     +                |
     +       -----------------------------------------
108  ++++++++| DPSF(DAY) =DPSE(IDAY1)                |
     +       -----------------------------------------
     +                |
     +          +-------------+
     +          +  DO LOOP TO  +
     +++++++++++ STMT #   109 +
     +          + IDAY1 = 1,273,10
     +          +-------------+
     +                |
     +       -----------------------------------------
     +       | DAY   =IDAY1+IDAY2                    |
     +       -----------------------------------------
     +                |
     +                                      ---------------------------
     +            * * * *                   |                         |
     +         *     IF     *    . T        |    GO TO 109            |
     +       *   DAY  .GT.      *+++++      |                         |
     +       * 273            *             |                         |
     +          *         *                 ---------------------------
     +            *  *  *
     +             F |
     +       -----------------------------------------
     +       | DPSE(DAY) =DPSE(IDAY1)                |
     +       -----------------------------------------
     +                |
109  ++++++++CONTINUE
                      |
             -----------------------------------------
             | DPSE(366) =DPSE(59)                   |
             | DPSE(365) =DPSE(363)                  |
             | DPSE(364) =DPSE(363)                  |
             -----------------------------------------
                      |
             CALL READ(BDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,XDNFS,FFOR,FFSI,MRNSM,
               DSMGH,PXCSA)
110          CALL READ(RMPF)
             CALL READ (RGPMB,AREA,FIMP,FWTR)
             CALL READ (VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,BIVF)
             CALL READ (OFSS,OFSL,OFMN,OFMNIS,IFRC)
             CALL READ (CSRX,FSRX,CHCAP,EXQPV,BFNLR,BFRC)
                      |
             -----------------------------------------
             | BFHRC=BFRC**(1.0/24.0)                |
             | BFRL =-ALOG(BFHRC)                    |
             | BFNRL=0.0                             |
             -----------------------------------------
                      |
                                            ---------------------------
                  * * * *                   |                         |
               *     IF     *    . T        |    GO TO 111            |
             *    BFNLR .L      *+++++      |                         |
             *T. 0.00001 .OR. BFN*          |                         |
             *LR .GT. 0.9999    *           ---------------------------
                *         *
                  *  *  *
                   F |
```

```
BFNHR=BFNLR**(1.0/24.0)
BFNRL=-ALOG(BFNHR)

111   IFPRC=IFRC**(1.0/96.0)
      IFRL =-ALOG(IFPRC)
      LSHFT=.FALSE.

      IF CONOPT(13) .NE. 1     T   GO TO 113
      F

      NBTRI=NCTRI
      FNTRI=NCTRI
      MXTRI=(10.0**EXQPV)*FNTRI+0.5

      IF MXTRI .GE. 98         T   WRITE(6,1)
      F                            FORMAT    1 FORMAT(29HWARNING:
                                             EXQPV ARRAY OVER RUN)

      NCSTRI =99

      DO LOOP TO
      STMT #   112
      KIA = 1, NBTRI

112   BTRI(KIA) =CTRI(KIA)
      TFCFS=1.0

      CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
        TFCFS)

113   EPAET=0.0
      FPER =1.0-FIMP-FWTR

      IF FPER .GT 0.01         T   GO TO 114
      F

      TPLR =100.0
      FPER =0.01

      115
```

```
114      TPLR =(1.0-FWTR)/FPER

115      VINTCR =0.25*VINTMR
         HSE  =0.0
         NRTRI=0
         SPIF =0.0
         CBF  =GWS*BFRL*(1.0+BFNRL*BFNX)
         SPDR =0.0
         OFUS =0.0

         OFUSIS =0.0
         OFR  =0.0
         OFRIS=0.0
         PEIS =0.0
         RHFO =0.0
         URHF =0.0
         AMIF =0.0
         AMNET=0.0
         AMPET=0.0
         AMSNE=0.0
         AMFSIL =0.0
         SASFX=0.0
         SARAX=0.0
         SRX  =CSRX
         VWIN =26.8888*AREA
         WCFS =24.0*VWIN
         RHFMC=0.025/WCFS
         TFCFS=CBF*WCFS
         SSRT =SQRT(OFSS)
         OFRF =1020.0*SSRT/(OFMN*OFSL)
         OFRFIS =1020.0*SSRT/(OFMNIS*OFSL)
         EQDF =0.00982*((OFMN*OFSL/SSRT)
              **0.6)
         EQDFIS =0.00982*((OFMNIS*OFSL/SSRT)
              **0.6)
         SOFRF=OFRF
         SOFRFI =OFRFIS
         SDEPTH =0.0
         IYR  =0
         MI   =0
         NI   =0
         MULT =0
         MULTI=CONOPT(10)
         ID   =1

         WRITE(6,9990)
           MULTI,  CONOPT(10),  CONOPT(15),  IBFLAG, ID

                                      FORMAT 9990  FORMAT(/3X,'
                                                    KWMAIN=  ',I5)
                                                    MULTI=  ',I5,'
                                                    CONOPT(10)=  ',I5,
         '  CONOPT(15)=  ',I5,'    IBFLAG=  ',I5,'

         ASM  =0.0

            IF
          CONOPT(7        T
         ) .EQ. 0       *****   GO TO 116
             F

         WT4AM=60.0
         WT4PM=60.0
         SAX  =15.0
         TANSM=0.0
         SPTW =0.0
         STMD =0.7
         SFMD =0.7
         ASMRG=0.0
```

```
                          |
116           CCNTINUE
         C  BEGIN NEW YEAR
                          |
             ------------------------------------------
117         | BYLZS=LZS                                |
            | BYUZS=UZS                                |
            | BYGWS=GWS                                |
            | BYIFS=IFS                                |
             ------------------------------------------
                          |
              +----------------+
              + DO LOOP TO     +
    ++++++++++ STMT #     118  +
    +         + KIA = 1,22     +
    +         +----------------+
    +                     |
    +        ------------------------------------------
    +       | CRFMI(KIA) =0.0                          |
    +       | SESF(KIA)  =0.0                          |
    +       | SERR(KIA)  =0.0                          |
    +       | SERA(KIA)  =0.0                          |
    +        ------------------------------------------
    +                     |
    +        ------------------------------------------
118 ++++++++| SQER(KIA) =0.0                           |
            | RGPM =RGPMB                              |
             ------------------------------------------
                          |
              +----------------+
              + DO LOOP TO     +
    ++++++++++ STMT #     119  +
    +         + KIA = 1,21     +
    +         +----------------+
    +                     |
    +        ------------------------------------------
    +       | T200FH(KIA) =0.0                         |
    +        ------------------------------------------
    +                     |
    +        ------------------------------------------
119 ++++++++| T20PRH(KIA) =0.0                         |
             ------------------------------------------
                          |
              +----------------+
              + DO LOOP TO     +
    ++++++++++ STMT #     120  +
    +         + KIA = 1,12     +
    +         +----------------+
    +                     |
    +        ------------------------------------------
120 ++++++++| EPCM(KIA) =1.0                           |
            | RDPT =0.0                                |
            | PDAY =274                                |
             ------------------------------------------
                          |
903         CALL READ(YR1,YR2)
                          |
             ------------------------------------------
            | TODARY(1,IYR) =YR2                       |
            | TCMARY(1,IYR) =YR2                       |
            | DPY  =365                                |
             ------------------------------------------
                          |
                 . * * * .            -------------------------
            . *     IF     * .  T    |     DPY = 366           |
          *   MOD(YR2,        *++++  |                         |
          *4) .EQ. 0          *      |                         |
              * .        . *          -------------------------
                   *  *
                 F |
                   |++++++++++++++++++++++++++++
                   |
                 . * * * .            -------------------------
            . *     IF     * .  T    |  CALL READ(CDSDR,NDSDR) |
          *   CONOPT(1        *++++  |                         |
          *)  .EQ. 1          *      |                         |
              * .        . *          -------------------------
                   *  *                         *
                 F |
```

```
                 |++++++++++++++++++++++++++++++
     ---------------------------------------------
     | NDSDP=0                                    |
     | MEDWY(5) =59                               |
     ---------------------------------------------
                 |
                 .                    ------------------------------------
           . *  * .                   |                                  |
      . *    IF      * .    T         |   MEDWY(5) = 366                 |
    *   DPY  .EQ.         *++++       |                                  |
    * 366                 *           |                                  |
      *  .      .    *                ------------------------------------
           *  .  *                                                   +
          F    |+++++++++++++++++++++++++++++++++++++++++++++++++++++++
 C  READ EVAPORATION DATA
                 |
                 .                    ------------------------------------
           . *  * .                   |                                  |
      . *    IF      * .    T         |   GO TO 125                      |
    *   CONOPT(           *++++       |                                  |
    *3) .NE. 1            *           |                                  |
      *  .      .    *                ------------------------------------
           *  .  *
          F    |
               |
          +-----------------+
          + DO LOOP TO      +
+++++++++++ STMT #     121  +
+         + KRD = 274,360,10
+         +-----------------+
+                  |
+
121 ++++++++CALL READ(DPET(KRD))
                   |
          +-----------------+
          + DO LOOP TO      +
+++++++++++ STMT #     122  +
+         + KRD = 1,273,10  +
+         +-----------------+
+                  |
+
122 ++++++++CALL READ(DPET(KRD))
                   |
          +-----------------+
          + DO LOOP TO      +
+++++++++++ STMT #     124  +
+         + IDAY2 = 1,9     +
+         +-----------------+
+                  |
+                  |
+                  |
+         +-----------------+
+         + DO LOOP TO      +
+++++++++++ STMT #     123  +
+.        + IDAY1 = 274,360,10
+         +-----------------+
+                  |
+     ---------------------------------------------
+     | DAY   =IDAY1+IDAY2                        |
+     ---------------------------------------------
+                  |
+     ---------------------------------------------
123 ++++++++| DPET(DAY) =DPET(IDAY1)              |
+     ---------------------------------------------
+                  |
+         +-----------------+
+         + DO LOOP TO      +
+++++++++++ STMT #     124  +
+         + IDAY1 = 1,273,10
+         +-----------------+
+                  |
+     ---------------------------------------------
+     | DAY   =IDAY1+IDAY2                        |
+     ---------------------------------------------
+                  |
+
```

```
          IF
    DAY .GT.                T   GO TO 124
    273
          F
| DPET(DAY) =DPET(IDAY1) |

124 ++++++++CONTINUE

| DPET(366) =DPET(59)  |
| DPET(365) =DPET(363) |
| DPET(364) =DPET(363) |

|127 |

125
          IF
    CONOPT(
    3) .EQ. 2               T   GO TO 130
          F
| DAY  =274 |

126    CALL READ (DPET(DAY))
          IF
    DAY .EQ.                T   GO TO 127
    273
          F
CALL DAYNXT(DAY, DPY)

|126 |

127        +  DO LOOP TO   +
++++++++++ +  STMT #   128 +
           +  MONTH = 1,12 +

128 ++++++++CALL READ(EPCM(MONTH))
          IF
    EPAET .N                T   GO TO 133
    E. 0.0
          F
           +  DO LOOP TO  +
++++++++++ +  STMT #  129 +
           +  DAY = 1,DPY +
```

```
      +                  |
      +                  |
      +       ------------------------------------------------------------
129   ++++++++| EPAET=EPAET+DPET(DAY)                                      |
              ------------------------------------------------------------
                         |
                         .                       ------------------------------------------
                 .  *  *  *  .                  |                                          |
              .  *     IF      *  .     T       |      EPAET  =  0.7*EPAET                 |
            *       EPCM(6)          *++++++    |                                          |
            *.NE. 1.0                *          |                                          |
              *  .            .  *               ------------------------------------------
                 *  .  *                                                                  +
                    F  |++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                       |
                   ---------
                   |131    |
                   ---------


130           CALL READ(EPAET,MNRD)
              ------------------------------------------------------------
              | EMAET=EPAET*(365.0+MNRD)/404.0                             |
              ------------------------------------------------------------
                         |
              CALL EVPDAY(DPET,EMAET)
                         |
              ------------------------------------------------------------
131           | AETX =24.0*EPAET/365.0                                     |
              | AEX96=1.2*AETX                                             |
              | AEX90=0.3*AETX                                             |
              | SIAM =1.2**SIAC                                            |
              | UZC   =SUZC*AEX90+BUZC*EXP(-2.7                            |
              |         *LZS/LZC)                                          |
              ------------------------------------------------------------
                         |
                         .                       ------------------------------------------
                 .  *  *  *  .                  |                                          |
              .  *     IF      *  .     T       |      UZC  =  0.25                        |
            *       UZC  .LT.        *++++++    |                                          |
            * 0.25                   *          |                                          |
              *  .            .  *               ------------------------------------------
                 *  .  *                                                                  +
                    F  |++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                       |
              ------------------------------------------------------------
              | SGRT =0                                                    |
              ------------------------------------------------------------
                       |
                +-----------------+
                +  DO LOOP TO     +
     ++++++++++++  STMT #    132  +
     +          +  DAY  =  1,366  +
     +          +-----------------+
     +                 |
     +        ------------------------------------------------------------
     +        | DDIW(DAY) =0.0                                             |
     +        | DRSF(DAY) =0.0                                             |
     +        | DRGPM(DAY) =RGPMB                                          |
     +        | DRSGP(DAY) =0.0                                            |
     +        ------------------------------------------------------------
     +                 |
     +          +-----------------+
     +          +  DO LOOP TO     +
     ++++++++++++  STMT #    132  +
     +          +  HOUR  =  1,24  +
     +          +-----------------+
     +                 |
132  ++++++++DRHP(DAY,HOUR) = 0.0
                       |
133                    .                       ------------------------------------------
                 .  *  *  *  .                  |                                          |
              .  *     IF      *  .     T       |      GO TO 135                           |
            *       CONOPT(9         *++++++    |                                          |
            *) .NE. 1                *          |                                          |
              *  .            .  *               ------------------------------------------
                 *  .  *
                    F  |
```

```
                          |
           --------------------------------------------
           | DAY   =274                               |
           | DRSF(366) =0.0                           |
           --------------------------------------------
                          |
134        CALL READ(DRSF(DAY))
           CALL DAYNXT(DAY, DPY)
                          |
                 . * . * .              ---------------------------
             . *    IF     * .   T      |      GO TO 134          |
           *    DAY .NE.         *+++++ |                         |
           * 274                 *      |                         |
             *  .          . *          ---------------------------
                   * . *
                   F  |
135                   |
                 . * . * .              ---------------------------
             . *    IF     * .   T      |      GO TO 137          |
           *    CONOPT(1         *+++++ |                         |
           *1)  .NE. 1           *      |                         |
             * .           . *          ---------------------------
                   * . *
                   F  |
           --------------------------------------------
           | DAY   =274                               |
           | DDIW(366) =0.0                           |
           --------------------------------------------
                          |
136        CALL READ(DDIW(DAY))
           CALL DAYNXT(DAY, DPY)
                          |
                 . * . * .              ---------------------------
             . *    IF     * .   T      |      GO TO 136          |
           *    DAY .NE.         *+++++ |                         |
           * 274                 *      |                         |
             *  .          . *          ---------------------------
                   * . *
                   F  |
137                   |
                 . * . * .              ---------------------------
             . *    IF     * .   T      |      GO TO 139          |
           *    CONOPT(7         *+++++ |                         |
           *)  .EQ. 0            *      |                         |
             * .           . *          ---------------------------
                   * . *
                   F  |
           --------------------------------------------
           | DAY   =274                               |
           --------------------------------------------
                          |
138        CALL READ(DMXT(DAY), DMNT(DAY))
           CALL DAYNXT(DAY, DPY)
                          |
                 . * . * .              ---------------------------
             . *    IF     * .   T      |      GO TO 138          |
           *    DAY .NE.         *+++++ |                         |
           * 274                 *      |                         |
             *  .          . *          ---------------------------
                   * . *
                   F  |
139        CALL READ(NSGRD)
                          |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                    I
                  * * *
              . *   IF    * .   T    ------------------------------
           *     NSGRD .E      *++++|     GO TO 141               |
           *Q. 0                *    |                             |
              *  .           . *     ------------------------------
                  * . *
                  F I

         CALL READ(WSG,SGRT)
                    I
                  * * *
              . *   IF    * .   T    ------------------------------
           *     CONOPT(        *++++|     CALL READ(WSG2,SGRT2,SG |
           *8)  .EQ. 1          *    |  MD)                        |
              *  .           . *     ------------------------------
                  * . *                                  +
                  F I++++++++++++++++++++++++++++++++++++++
                    I
              +--------------+
              + DO LOOP TO   +
    ++++++++++ STMT #    140 +
    +         + KRD = 1,NSGRD+
    +         +--------------+
    +               I
    +
    +    CALL READ(ISGRD)
    +
140 ++++++++CALL READ(DRSGP(ISGRD))
      C  READ RECORDING RAIN GAGE HOURLY TOTALS

141      CALL READ(IWBG,YEAR,MONTH,DATE,CN)
      C  PUNCH NO NUMBER AFTER CN ON YEAR .EQ. 98 CARD
                    I
                  * * *
              . *   IF    * .   T    ------------------------------
           *     YEAR .GE       *++++|     GO TO 144               |
           *. 98                 *   |                             |
              *  .           . *     ------------------------------
                  * . *
                  F I
                    I
         ------------------------------------------------
         | HRF  =12*(CN-1)+1                            |
         | HRL  =12*(CN-1)+12                           |
         | DAY  =MEDCY(MONTH)+DATE                      |
         ------------------------------------------------
                    I
              +--------------+
              + DO LOOP TO   +
    ++++++++++ STMT #    142 +
    +         + HOUR = HRF, HRL
    +         +--------------+
    +               I
    +
142 ++++++++CALL READ(DRHP(DAY,HOUR))
                    I
                  * * *
              . *   IF    * .   T    ------------------------------
           *     DPY .NE.       *++++|     GO TO 141               |
           * 366 .OR. MONTH .NE*     |                             |
           *. 2 .OR. DATE .NE. *     |                             |
              *  .           . *     ------------------------------
                  * . *
                  F I
              +--------------+
              + DO LOOP TO   +
    ++++++++++ STMT #    143 +
    +         + HOUR = HRF, HRL
    +         +--------------+
    +               I
    +
    +    DRHP(366,HOUR) = DRHP(60,HOUR)
    +
143 ++++++++DRHP(60,HOUR) = 0.0
                    I
```

```
              |141  |

         C  CALCULATE PRECIPITATION WEIGHTING FACTORS

144         | DAY   =274                                       |

                    IF                 T
              NSGRD  .E           *++++   GO TO 151
            *Q. 0
                    F

            | PDAY =274                                        |
            | RDPT =0.0                                        |

145         | EHSGD=SGRT                                       |

                    IF                 T
              SGRT  .EQ           *++++   EHSGD  = 24
            *. 0
                    F

            | EHSGDF =EHSGD                                    |

146         CONTINUE

                +  DO LOOP TO   +
       +++++++++ STMT #    150  +
                +  HOUR = 1,24  +

            | RDPT =RDPT+DRHP(DAY,HOUR)                        |

                    IF                 T
              HOUR  .NE           *++++   GO TO 150
            *. EHSGD
                    F

                    IF                 T
              RDPT  .LE           *++++   GO TO 147
            *. 0.0
                    F
```

```
         IF                 T
    SGRT .EQ            ++++    PDAY = DAY
  0
       F
         ++++++++++++++++++++++++++++++

  DRGPM(PDAY) =(DRSGP(DAY)*WSG+RDPT
         *(1.0-WSG))/RDPT

         IF                 T
    CONOPT(3            ++++    DPET(PDAY) = 0.5*DPET(P
  ) .NE. 0                      DAY)
       F
         ++++++++++++++++++++++++++++++

         IF                 T
    SGRT .NE            ++++    PDAY = DAY
  0
       F
         ++++++++++++++++++++++++++++++

  RDPT =0.0

  150

147
         IF                 T
    DRSGP(DA            ++++    GO TO 149
  Y) .LE. 0.0
       F

      DO LOOP TO
      STMT #    148
      KHOUR = 1,EHSGD

148 +++++++DRHP(DAY,KHOUR) = (WSG*DRSGP(DAY))/EHSGDF
149
         IF                 T
    SGRT .NE            ++++    PDAY = DAY
  0
       F
         ++++++++++++++++++++++++++++++
150 +++++++CONTINUE
       CALL DAYNXT(DAY,DPY)

         IF                 T
    DAY .EQ.            ++++    GO TO 151
  274
       F
```

C-3

ORIGINAL PAGE IS
OF POOR QUALITY

```
                    |
               . * . * .
          . *     IF      * .    T   ----------------------------------------
        *    CONOPT(8         * ++++|     GO TO 146                          |
        *)  .EQ. 0           *      |                                        |
          * .            . *         ----------------------------------------
               * . *
               F  |
                  |
               . * . * .
          . *     IF      * .    T   ----------------------------------------
        *     DAY .NE.        * ++++|     GO TO 146                          |
        *  SGMD              *.     |                                        |
          * .            . *         ----------------------------------------
               * . *
               F  |
                  |
     ---------------------------------------------
     | WSG   =WSG2                               |
     | SGRT  =SGRT2                              |
     ---------------------------------------------
                  |
               -------
               |145  |
               -------

151         BACKSPACE 11
                  |
      +--------------------------------------------+
     +  WRITE (11)                                +
    +   (RPLCTC(I),I=1,1832)                     +
   +                                            +
  +--------------------------------------------+
                  |
            BACKSPACE 18
                  |
      +--------------------------------------------+
     +  WRITE (18)                                +
    +   (RCOMMA(I),I=1,12387)                    +
   +                                            +
  +--------------------------------------------+
                  |
            RETURN
END
```

```
SUBROUTINE WORSTC(PCAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)

      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
      KFLAG,IDFLAG,
      IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
        NCTRI,CTRI,FIRR,RICY,DPSE,BODFSM,SPBFLW,SPTWCC,SPM,ELDIF,
        XCNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
       SATRI,UHFA,
      MNRD,
        FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
        BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,
        BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
        IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAET,FPER,
        TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPDR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
        RHFC,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
        WCFS,RHFMC,SSRT,JFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
        SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
      DEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
      TCMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
      TSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
      TORARY(1,6,1),
      DRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)

      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
      TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,FDLZS,
      EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
      TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,
      TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
      TRHF,
      SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS

C     KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)

C       BASED ON STANFORD WATERSHED MODELS III & IV

      DIMENSION BTRI(99),CONOPT(16),CRFMI(22),CTRI(99),DDIW(366),
        DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
        DRSGP(366), DPET(366), DRSF(366), DSSF(366), FDLZS(366),
       EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
       EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
        RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),
       THSF(24),TITLE(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
       TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
       TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T200FH(21),T20PRH(21),
       UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
       PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)

      LOGICAL  LSHFT

      INTEGER  CDSDR,CN,CONOPT,DATE,DAY,DPY,EHSGD,HOUR,HRF,HRL,PDAY,
        PRD,RHPD,RHPH,RSBC,SGMD,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX

      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY

      INTEGER  DATES,EDATE

      DIMENSION RPLOTC(1832),RCOMMA(12087)

      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))

      REAL  IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT

      DATA  MEDCY/  0,      31,59,90,120,151,181,212,243,273,304,334/

      DATA  MEDWY/304,334,365,31,59,90,120,151,181,212,243,273        /

      REAL     MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD

      REAL  SSQM,SSQDI,SSQMI,VDRSF,VDSSF

      REAL  VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
           SMSQM

      REAL  MDRSF,MDSSF,MMRSF,MMSSF

      BACKSPACE 11

      READ  (11)
      (RPLOTC(I),I=1,1832)
```

```
            BACKSPACE 18

            READ(18)
            (RCOMMA(I),I=1,12087)

150         CONTINUE

            DATE =PDATE
            DAY  =DATES
            MDAY =PDAY
            SINDEX =0
            MAXI =0
            AMRPM=0.0
            AMPREC =0.0
            AMBF =0.0
            AMSE =0.0
            AMSTF=0.0
            AMRTF=0.0

            IF
        DPY .EQ.          T    MEDWY(5)=366
        366
            F

            WRITE(6,3)
            (TITLE(KTA),KTA=1,18)

                                   FORMAT    3 FORMAT (1H1,10X,18A4,//)

            WRITE(6,4)
            MSBDIC

                                   FORMAT    4 FORMAT(1X,'WORSE CASE
                                               HOURLY CFS VALUES',//,1X,
        C  BEGIN DAY LOOP                      4A4)

            IMONTH =MONTH

152         TDSF =0.0

            IF
        MONTH.NE          T    GO TO 148
        .4
            F

            IF
        MDAY.NE.          T    GO TO 149
        31
            F
```

```
148        IF DATE.GT.(MOD(DAY,MDAY))    T ---->  IMONTH=MONTH+1
           F

149        CONTINUE

           PET  =ERCM(IMONTH)*UPET(DAY)
           PETU =PET
           TFMAX=0.0

      C  EVAPOTRANSPIRATION ADJUSTMENTS

           IF CONOPT(7) .NE. 1           T ---->  GO TO 153
           F

           IF DMXT(DAY) - 4.0*ELDIF .LT. 40.0    T ---->  PET = 0.0
           F

           IF SPTW .GT. SPTWCC           T ---->  PET = FFOR*PET
           F

      C  CALCULATION OF SNOW EVAPORATION

           IF DMNT(DAY) .GT. 32.0 .OR. SPTW .LE. DPSE(DAY)    T ---->  GO TO 153
           F

           SE   =DPSE(DAY)
           AMSNE=AMSNE+SE
           SPTW =SPTW-SE

           IF SFMD .GT. 0.0              T ---->  SDEPTH = SDEPTH - SE/SFMD
           F
```

```
                   +-----!-----+
153                + DO LOOP TO +
     ++++++++++++  + STMT #  202 +
     +             + HOUR = 1,24 +
     +             +-----------+
     +                   !
     +(NSGRD IF((NSGRD .EQ. 0) .AND. (DRHP(DAY,HOUR) .NE. 0.0)  .AND. (PET .EQ.
     +         PETU) .AND. (CONOPT(3) .EQ. 1)) PET = 0.5*PET
     +                   !
154  +                              ------------------------------
     +               IF         T  |    RGPM = DRGPM(DAY)         |
     +          HOUR .EQ       *++++|                              |
     +        SGRT + 1             |                              |
     +                              ------------------------------
     +                 F |+++++++++++++++++++++++++++++++
     +
     + (CONTINUED ON PAGE   5)

                                                         PAGE   5
     +
     +                   !
     +                              ------------------------------
     +               IF         T  |    HSE = (FWTR*PET)/12.0     |
     +          HOUR .EQ       *++++|                              |
     +        9                    |                              |
     +                              ------------------------------
     +                 F |+++++++++++++++++++++++++++++++
     +
     +                              ------------------------------
     +               IF         T  |    HSE = 0.0                 |
     +          HOUR .EQ       *++++|                              |
     +        .21                  |                              |
     +                              ------------------------------
     +                 F |+++++++++++++++++++++++++++++++
     +
     +       -----------------------------------------
     +      | PRH   =1.2*DRHP(DAY,HOUR)               |
     +      | AMPREC =AMPREC+PRH                      |
     +       -----------------------------------------
     +                   !
     +  C  ENTER SNOWMELT SUBROUTINE
     +                   !
     +                              ------------------------------
     +               IF         T  |    CALL SNOMEL(BDDFSM,SPTW   |
     +          CONOPT(7       *++++| CC,SPM,ELDIF,DAY,           |
     +        ) .EQ. 1             |                              |
     +                              ------------------------------
     +                 F |+++++++++++++++++++++++++++++++
     +      SPBFLW, XDNFS,FFOR,FFSI,MRNSM,DSMGH,SDEPTH,STMD, PXCSA,HOUR,
     +      SAX,SOFRF,OFRFIS,SOFRFI,AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,
     +      WT4AM,WT4PM,ASM,ASMRG,SASFX,SARAX,DMXT,DMNT,RICY,FIRR)
     +
     +       -----------------------------------------
155  +      | AMRPM=AMRPM+PRH                         |
     +       -----------------------------------------
     +                   !
     +       -----------------------------------------
156  +      | TOFR =0.0                               |
     +      | ARHF =0.0                               |
     +       -----------------------------------------
     +                   !
     +  C  15 MINUTE ACCOUNTING AND ROUTING LOOP
     +                   !
     +             +-----------+
     +             + DO LOOP TO +
     ++++++++++++  + STMT #  187 +
     +             + PRD = 1,4  +
     +             +-----!-----+
     +                   !
```

```
PEBI  =0.0
PPI   =0.0
OFR   =0.0
OFRIS=0.0
WI    =0.0
WEIFS=0.0
PMEUZS =0.0
PMELZS =0.0
PMEIFS =0.0
PMECFS =0.0

PEP   =0.25*PRH

IF CONOPT(2) .EQ. 1    T  CALL PREPRD(RGPM,DRHP,DAY,HOUR,DPY,PRD,PEP,PRH)
F

IF PEP .GT. 0.0        T  GO TO 157
F

IF OFUS .GT. 0.0       T  GO TO 159
F

IF IFS .GT. 0.0        T  GO TO 170
F

IF NRTRI .GT. 0        T  GO TO 172
F

TRHF =0.0

IF RHFO .GT. 0.0       T  GO TO 181
F
```

```
           |
       ----------
       |184    |
       ----------

C  RAINFALL UPPER ZONE INTERACTION
             |
157          .
        . *     * .
     . *   IF      *  .   T    -------------------------------
   *     PEP  .GE.         *++++|   GO TO 158                  |
   * VINTCR                *    |                              |
        * .       . *           -------------------------------
           *  .  *
           F  |
 -----------------------------------------
 | UZS    =UZS+PEP*TPLR                   |
 | VINTCR =VINTCR-PEP                     |
 | PPI    =0.0                            |
 | PEBI   =0.0                            |
 | PMEUZS =PEP                            |
 -----------------------------------------
              |
              .
        . *     * .
     . *   IF      *  .   T    -------------------------------
   *     OFUS  .GT             *++++|   GO TO 159              |
   *.  0.0                 *    |                              |
        * .       . *           -------------------------------
           *  .  *
           F  |
         ----------
         |170    |
         ----------

              |
 -----------------------------------------
158 | PPI    =PEP-VINTCR                  |
 | UZS    =UZS+VINTCR*TPLR                |
 | VINTCR =0.0                            |
 | LZSR   =LZS/LZC                        |
 | UZC    =SUZC*AEX90+BUZC*EXP(-2.7       |
 |         *LZSR)                         |
 -----------------------------------------
              |
              .
        . *     * .
     . *   IF      *  .   T    -------------------------------
   *     UZC  .LT.         *++++|   UZC = 0.25                 |
   * 0.25                  *    |                              |
        * .       . *           -------------------------------
           *  .  *
           F  |                                   +
              |++++++++++++++++++++++++++++++++++++
 -----------------------------------------
 | UZRX =2.0*ABS(UZS/UZC-1.0)+1.0         |
 | FMR  =(1.0/(1.0+UZRX))**UZRX           |
 -----------------------------------------
              |
              .
        . *     * .
     . *   IF      *  .   T    -------------------------------
   *     UZS  .GT.         *++++|   FMR = 1.0 - FMR            |
   * UZC                   *    |                              |
        * .       . *           -------------------------------
           *  .  *
           F  |                                   +
              |++++++++++++++++++++++++++++++++++++
 -----------------------------------------
 | PEBI   =PPI*FMR                        |
 | PMEUZS =PEP-PEBI                       |
 | UZS    =UZS+PPI-PEBI                   |
 -----------------------------------------
```

```
     C  LOWER ZONE AND GROUNDWATER INFILTRATION

159     | LZSR =LZS/LZC                               |
        | EID  =4.0*LZSR                              |

        IF LZSR .LE 1.0          T  |  GO TO 160  |
        F

        | EID  =4.0+2.0*(LZSR-1.0)                    |

        IF LZSR .LE 2.0          T  |  GO TO 160  |
        F

        | EID  =6.0                                   |

160     | PEBI =PEBI+OFUS                             |
        | CMIR =0.25*SIAM*BMIR/(2.0**EID)             |
        | CIVM =BIVF*2.0**LZSR                        |

        IF CIVM .LT 1.0          T  |  CIVM = 1.0  |
        F

        | PEAI =PEBI*PEBI/(2.0*CMIR*CIVM)             |

        | WI   =PEBI*PEBI/(2.0*CMIR)                  |

        IF PEBI .GE CMIR         T  |  WI = PEBI - 0.5*CMIR  |
        F

        IF PEBI .GE CMIR*CIVM    T  |  PEAI = PEBI - 0.5*CMIR*CIVM  |
        F

        | WEIFS=WI-PEAI                               |
```

```
          IF PEBI .LE. OFUS                    T --> GO TO 161
          F
     PMELZS =(PEBI-WI)*((PEBI-OFUS)
          /PEBI)
     PMEIFS =WEIFS*((PEBI-OFUS)/PEBI)
     PMECFS =PEAI*((PEBI-OFUS)/PEBI)

161  CONTINUE

          IF (PEAI -OFUS) .GT. 0.0             T --> GO TO 162
          F
     EQD  =(OFUS+PEAI)/2.0

          163

162  ECD  =EQDF*((PEAI-OFUS)**0.6)

163       IF (OFUS + PEAI) .GT. (2.0*EQD)      T --> EQD = 0.5*(OFUS + PEAI)
          F

          IF (OFUS + PEAI) .LE. 0.001          T --> GO TO 164
          F
     OFR  =0.25*OFRF*(((OFUS+PEAI)*0.5)
          **1.67)*((1.0+0.6*((OFUS
          +
          PEAI)/(2.0*EQD))**3.0)**1.67)

          IF OFR .GT. (0.75*PEAI)              T --> OFR = 0.75*PEAI
          F
```

```
164 +        . * IF * .  T      GO TO 168
         *  FIMP .EQ    *++++
         *. 0.0         *
              F |
165 +    | PEIS =PPI+OFUSIS                     |
                |
         . * IF * .  T      GO TO 166
         *  (PEIS -     *++++
         *OFUSIS) .GT. 0.0  *
              F |
         | EQDIS=(OFUSIS+PEIS)/2.0              |
                |
         -- --|167 |

                |
166 +    | EQDIS=EQDFIS*((PEIS-OFUSIS)**0.6)    |
                |
167 +    . * IF * .  T      EQDIS = 0.5*(OFUSIS + P
         * - - (OFUSIS  *++++  EIS)
         *+ PEIS) .GT. (2.0*E*
         *QDIS)         *
              F |++++++++++++++++++++++++++++++++
         . * IF * .  T      GO TO 168
         *  (OFUSIS     *++++
         *+ PEIS) .LE. 0.01  *
              F |
         | OFRIS=0.25*OFRFIS*(((OFUSIS+PEIS)
         |       *0.5)**1.67)*((1.0+0.6*((
         |          OFUSIS + PEIS)/(2.0*EQDFIS))**3.0)**1.67)
                |
         . * IF * .  T      OFRIS = PEIS
         *  OFRIS .G    *++++
         *T. PEIS       *
              F |++++++++++++++++++++++++++++++++
168 +    | TOFR =TOFR+FPER*OFR+FIMP*OFRIS
         |       +PPI*FWTR
         | OFUSIS =PEIS-OFRIS
         | OFUS =PEAI-OFR
                |
```

```
            IF                T     GO TO 169
       OFUS .GE          *****
   0.001
            F

 LZS  =LZS+OFUS
 OFUS =0.0
 CFRIS=CFRIS+OFUSIS
 OFUSIS =0.0

169
 LZRX =1.5*ABS(LZS/LZC-1.0)+1.0
 FMR  =(1.0/(1.0+LZRX))**LZRX

            IF                T     FMR = 1.0 - FMR*(LZS/LZ
       LZS .LT.          *****     C)
   LZC
            F

 PLZS =FMR*(PEBI-WI)
 PGW  =(1.0-FMR)*(PEBI-WI)*(1.0
        -SUBWF)*FPER
 GWS  =GWS+PGW
 BFNX =BFNX+PGW
 LZS  =LZS+PLZS
 IFS  =IFS+WEIFS*FPER

170
 SPIF =IFRL*IFS
 AMIF =AMIF+SPIF
 IFS  =IFS-SPIF

            IF                T     GO TO 171
       IFS .GE.          *****
   0.0001
            F

 LZS  =LZS+IFS
 IFS  =0.0

171
 UHFA(1)  =FPER*CFR+PPI*FWTR+FIMP
           *OFRIS+SPIF
 SPOR =UHFA(1)

C    ROUTING
172
            IF                T     GO TO 173
       CONOPT(1          *****
   2)  .NE. 1
            F

 URHF =URHF+0.25*UHFA(1)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
        IF                    T   GO TO 181
    PRD .NE.
  4
        F

    UHFA(1) =URHF

173 TRHF =0.0
    KTRI =NCTRI

        IF                    T   KTRI = NCSTRI
    CONOPT(1
  3) .EQ. 1
        F

174 URHF =UHFA(KTRI)

        IF                    T   GO TO 176
    URHF .LE
  0.0
        F

175 TRHF =TRHF+URHF*CTRI(KTRI)

        IF                    T   TRHF = TRHF +
    CONOPT(1                      URHF*SATRI(KTRI - 1)
  3) .EQ. 1 .AND. LSH             UHFA(KTRI + 1) = URHF
  FT .AND. KTRI .GE.              177
        F

176 UHFA(KTRI+ 1) = 0.0

177 KTRI =KTRI-1

        IF                    T   GO TO 174
    KTRI .GE
  1
        F
```

```
178  IF URHF .LE. 0.0  --T-->  GO TO 179
     F
     NRTRI=NCTRI
     IF CONOPT(13) .EQ. 1  --T-->  NRTRI = MXTRI
     F
179  NRTRI=NRTRI-1
     UHFA(1) =0.0
     IF CONOPT(13) .NE. 1  --T-->  GO TO 180
     F
     NNSTRI =NCSTRI+1
     UHFA(NNSTRI) =0.0
180  URHF =0.0
181  IF SRX .LE. CSRX  --T-->  SRX = CSRX
     F
     RHF1 =TRHF-SRX*(TRHF-RHF0)
     RHF0 =RHF1
     IF RHF0 .LT. RHFMC  --T-->  RHF0 = 0.0
     F
     TFCFS=(4.0*RHF1+CBF-HSE)*WCFS
```

```
                        IF                 T   GO TO 182
            CONOPT(1
        *3) .NE. 1
                        F

                        IF                 T   GO TO 182
            CONOPT(1
        *2) .EQ. 1 .AND. PRD*
        * .NE. 4
                        F

     CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
       TFCFS)

     DATE =MOD(DAY,MDAY)

                        IF                 T   WRITE(6,6) DATE,HOUR,PR
            LSHFT                              D,NCSTRI
                        F
                                    FORMAT     6 FORMAT(2X,I2,2X,I2,2X,
      8HELEMENTS)                              I2,2X,20HHISTOGRAM
182  CONTINUE                                  CHANGES TO,1X,I2,1X,

                        IF                 T   SRX = CSRX
            TFCFS .L
        *E. 0.5*CHCAP
                        F

                        IF                 T   SRX = CSRX
            (TFCFS .
        *GT. 0.5*CHCAP) .AND*
        *. (TFCFS .LT. 2.0*C*
                        F
      +(FSRX - CSRX)*((TFCFS - 0.5*CHCAP)/(1.5*CHCAP))**3

                        IF                 T   SRX = FSRX
            TFCFS .G
        *T. 2.0*CHCAP
                        F

                        IF                 T   GO TO 183
            TFCFS .L
        *E. TFMAX
                        F

     PRDF =PRD
     TOFP24 =HCUR
```

```
             IF              T    TDFP24 = (TDFP24 - 1.0)
        PRD .LE.                  + 0.15*PRDF
      3
             F
  | TFMAX=TFCFS                                    |

183
  | ARHF =ARHF+RHF1                                |

C   STCRM OUTPUT REQUESTED BY CONOPT(1)

184
             IF              T    GO TO 186
        CONOPT(1
      ) .NE. 1
             F

             IF              T    GO TO 186
        DAY .NE.
      CDSDR
             F

             IF              T    WRITE(6,7)
        HOUR .EQ
      1 .AND. PRD .EQ.
      1
             F
                                             FORMAT     7 FORMAT(1H//,21X,
                         19HRAINFALL DEPOSITION,12X,16HMOISTURE STORAGE,
   14X,17HSTREAMFLOW ORIGIN,6X,14HSTREAM OUTFLOW/2X,116HDY HR PD RA
  IN     EUZS    ELZS     EIFS     EOFS         UZS       LZS     IFS      OFS       S
  PCF     SPIF     SPBF     SPTF   INCHES      CFS)

  | DATE  =MOD(DAY,MDAY)                           |
  | OFS   =OFUS*FPER+OFUSIS*FIMP                   |
  | SPCF  =OFR*FPER+OFRIS*FIMP+PPI*FWTR            |
  | SPBF  =0.25*(CBF-HSE)                          |
  | SPTF  =SPDR+SPBF                               |
  | SPDR  =0.0                                     |

             IF              T    TFCFS = (CBF - HSE)*WCF
        RHFO .LE                  S
      . 0.0
             F

  | RSPTF=0.25*TFCFS/WCFS                          |

     WRITE(6,8)
     DATE,HOUR,PRD,PEP,PMEUZS,PMELZS,PMEIFS,PMEOFS,U+
   +ZS,LZS
```

```
     +     ,IFS,CFS,SPGF,SPIF,SPBF,SPTF,RSPIF,TFCFS
     +                                           FORMAT     8 FORMAT(2X,I2,
     +       F7.1)                               1X,I2,1X,I1,5(1X,F6.4),2X,
     +                                           4(F7.4),2X,5(1X,F6.4),1X,
     +          . * * .
     +     . *    IF      * .   T     GO TO 185
     +     *   HOUR  .EQ        *+++++
     +     *. 24  .AND.  PRD  .EQ.*
     +     * 4
     +          *  .  *
     +             F
     +
     +          |186  |
     +
185  +     | NDSDP=NDSDP+1                              |
     +
     +          . * * .
     +     . *    IF      * .   T     GO TO 186
     +     *   NDSDR  .E        *+++++
     +     *Q. NDSDP            *
     +          *  .  *
     +             F
     +
     +     CALL DAYNXT(CDSDR,DPY)
186  +     CONTINUE
     +          . * * .
     +     . *    IF      * .   T     VINTCR = VINTCR + DPET(
     +     *   VINTCR .         *+++++  DAY)/96.0
     +     *LT. 0.25*VINTMR     *
     +          *  .  *
     +             F  |+++++++++++++++++++++++++++++++++
187  +++++++++CONTINUE
     +   C  END OF 15 MINUTE LOOP
     +
     +          . * * .
     +     . *    IF      * .   T     GO TO 197
     +     *   CONOPT(5         *+++++
     +     *) .NE. 1            *
     +          *  .  *
     +             F
     +
     +   C  HOURLY OVERLAND FLOW AND RAINFALL SORTING
     +
     +          . * * .
     +     . *    IF      * .   T     GO TO 193
     +     *   TOFR  .LE        *+++++
     +     *. 0.0               *
     +          *  .  *
     +             F
     +
     +     | KT20 =20                                   |
     +
```

```
188     IF KT20 .LT. 1                T  ----> GO TO 192
        F
        IF TOFR .GT. T20CFH(KT20)     T  ----> GO TO 189
        F
        |190|

189     T20OFH(KT20+1) = T20OFH(KT20)
        |191|

190     T20OFH(KT20+1) = TOFR
        |193|

191     | KT20 =KT20-1 |
        |188|

192     | T20CFH(1) =TCFR |
193     IF PRH .LE. 0.0               T  ----> GO TO 197
        F
        | KT20 =20 |
194     IF KT20 .LT. 1                T  ----> GO TO 196
        F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
          |
      * . * . * .
   . *    IF     * .  T       ------------------------------
 *    PRH .GT.         *++++ |   GO TO 195                  |
 *  T20PRH(KT20)       *     |                              |
      * .       . *           ------------------------------
          * . *
          F |

   T20PRH(KT20 + 1) = PRH
          |
        ------
        |197 |
        ------


195    T20PRH(KT20+1) = T20PRH(KT20)
          |
   --------------------------------------------
   | KT20 =KT20-1                             |
   --------------------------------------------
          |
        ------
        |194 |
        ------


          |
   --------------------------------------------
196| T20PRH(1) =PRH                           |
   --------------------------------------------
          |
C  ADDING GROUNDWATER FLOW
          |
   --------------------------------------------
197| CBF  =GWS*BFRL*(1.0+BFNRL*BFNX)          |
   | GWS  =GWS-CBF                            |
   | AMBF =AMBF+CBF                           |
   | THGR =ARHF+CBF                           |
   --------------------------------------------
          |
      * . * . * .
   . *    IF     * .  T       ------------------------------
 *    HSE .GT.         *++++ |   HSE = THGR                 |
 * THGR                *     |                              |
      * .       . *           ------------------------------
          * . *                                     +
          F |+++++++++++++++++++++++++++++++++++++++
   --------------------------------------------
   | AMSE =AMSE+HSE                           |
   | THSF(HOUR) =(THGR-HSE)*WCFS              |
   | TDSF =TDSF+THSF(HOUR)                    |
   --------------------------------------------
          |
C

C STORE SIMULATED HOURLY STREAM FLOWS

C  DRAINING OF UPPER ZONE STORAGE
          |
   --------------------------------------------
   | SINDEX =SINDEX+1                         |
   --------------------------------------------
          |
   THSFD(SINDEX,1)=THSF(HOUR)
          |
   --------------------------------------------
306| UZINFX =(LZS/UZC)-(LZS/LZC)              |
   --------------------------------------------
          |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
IF UZINFX .LE. 0.0                  T  GO TO 198
F
LZSR =LZS/LZC
UZINLZ =0.003*BMIR*UZC*UZINFX*
      *3.0

IF UZINLZ .GT. UZS                  T  UZINLZ = UZS
F
UZS  =UZS-UZINLZ
LZRX =1.5*ABS(LZSR-1.0)+1.0
FMR  =(1.0/(1.0+LZRX))**LZRX

IF LZS .LT. LZC                     T  FMR = 1.0 - FMR*LZSR
F
PGW  =(1.0-FMR)*UZINLZ*(1.0-SUBWF)
      *FPER
PLZS =FMR*UZINLZ
LZS  =LZS+PLZS
GWS  =GWS+PGW
BFNX =BFNX+PGW

C   4 PM ADJUSTMENTS OF VARIOUS VALUES

198
IF HOUR .NE. 16                     T  GO TO 202
F
AEX90=0.9*(AEX90+PET)
AEX96=C.96*(AEX96+PET)

C   INFILTRATICN CORRECTICN

SIAM =(AEX96/AETX)**SIAC

IF SIAM .LT. 0.33                   T  SIAM = 0.33
F
```

```
      BFNX =0.97*BFNX

      IF PET .EQ. 0.0          T  →  GO TO 202
      F

C  EVAP-TRANS LOSS FRCM GROUNDWATER

      GWET =GWS*GWETF*PET*FPER
      GWS  =GWS-GWET
      BFNX =BFNX-GWET

      IF BFNX .LT. 0.0         T  →  BFNX = 0.0
      F

      AMPET=AMPET+PET

      IF PET .GE. UZS          T  →  GO TO 199
      F

      UZS  =UZS-PET
      AMNET=AMNET+PET

      202

199   PET  =PET-UZS
      AMNET=AMNET+UZS
      UZS  =0.0
      LZSR =LZS/LZC

      IF PET .GE. ETLF*LZSR    T  →  GO TO 200
      F

      SET  =PET*(1.0-PET/(2.0*ETLF*LZSR))

      201
```

```
200      | SET   =0.5*ETLF*LZSR                          |

201      | LZS   =LZS-SET                                |
         | AMNET=AMNET+SET                               |

202 ++++++++CONTINUE
     C  END OF HOUR LOOP

         | DSSF(DAY) =TDSF/24.0                          |

              IF                T        DSSF(DAY) = DSSF(DAY) +
          CONOPT(1            *+++++   DDIW(DAY)
     *1) .EQ. 1
            F

203      | AMRTF=AMRTF+DRSF(DAY)                         |
         | AMSTF=AMSTF+DSSF(DAY)                         |

              IF                T        EDLZS(DAY) = LZS
          CONOPT(6            *+++++
     *) .EQ. 1
            F

     C  STORE ERRORS AND FLOW DURATION

              IF                T        GO TO 204
          CONOPT(4            *+++++
     *) .NE. 1
            F

         | ERR   =DSSF(DAY)-DRSF(DAY)                    |

              IF                T        KRFMI = 1.0
          DRSF(DAY            *+++++
     *) .LT. 1.0
            F

              IF                T        KRFMI= 2.0*ALOG(DRSF(DA
          DRSF(DAY            *+++++   Y)) + 2.0
     *) .GT. 1.0
            F

         | CRFMI(KRFMI) =CRFMI(KRFMI)+1.0                |
         | SERR(KRFMI)  =SERR(KRFMI)+ERR                 |
         | SERA(KRFMI)  =SERA(KRFMI)+ABS(ERR)            |
         | SQER(KRFMI)  =SQER(KRFMI)+ERR*ERR             |
         | SESF(KRFMI)  =0.0                             |
```

```
                         |
                         .
                   .  *     *  .
              .  *     IF       *  .     T    -------------------------------------------
          *      CRFMI(KR                *++++|     SESF(KRFMI) = SQRT(ABS(              |
          *FMI)  .GT. 1.0                *    |   (SQER(KRFMI) -                         |
              *  .              .  *          |                                          |
                   *  .     .  *              -------------------------------------------
                   F     |                                         +
                         |+++++++++++++++++++++++++++++++++++++++++++
           SERR(KRFMI)**2/CRFMI(KRFMI))/(CRFMI(KRFMI) - 1.0)))
204             CONTINUE
                         |
               -----------------------------------------------
               | DATE =DAY                                   |
               -----------------------------------------------
                         |
                         |
                         .
                   .  *     *  .
              .  *     IF       *  .     T    -------------------------------------------
          *       (MCNTH.E               *++++|        DATE=MOD(DAY,MDAY)                |
          *Q.4).AND.(MCAY.EQ.3*          *    |                                          |
          *1)                            *    |                                          |
              *  .              .  *          -------------------------------------------
                   *  .     .  *                                   +
                   F     |+++++++++++++++++++++++++++++++++++++++++++
                         |
                         .
                   .  *     *  .
              .  *     IF       *  .     T    -------------------------------------------
          *        MONTH.NE              *++++|        DATE=MOD(DAY,MDAY)                |
          *.4                            *    |                                          |
              *  .              .  *          |                                          |
                   *  .     .  *              -------------------------------------------
                   F     |                                         +
                         |+++++++++++++++++++++++++++++++++++++++++++
              +--------------------------------------------------------+
            +   WRITE(6,9)                                            +
          +     DATE, (THSF(HCUR),HOUR=1,12)                        +
        +                                                         +
      +--------------------------------------------------------+
                         |
                         |                        FORMAT    9 FORMAT(1H/,1X/,1X,I4,2X,
                         |                                  2HAM,1X,6F8.1,3X,6F8.1)
              +--------------------------------------------------------+
            +   WRITE(6,10)                                           +
          +     (THSF(HOUR),HOUR=13,24), DSSF(DAY)                  +
        +                                                         +
      +--------------------------------------------------------+
                         |
                         |                        FORMAT   10 FORMAT(1HJ,6X,2HPM,1X,
                         |                                        6F8.1,3X,7F8.1)
                         .
                   .  *     *  .
              .  *     IF       *  .     T    -------------------------------------------
          *        TDFP24 .              *++++|      GO TO 205                           |
          *LT. 12.0                      *    |                                          |
              *  .              .  *          |                                          |
                   *  .     .  *              -------------------------------------------
                   F     |
                         |
               -----------------------------------------------
               | TDFP12 =TDFP24-12.0                         |
               -----------------------------------------------
                         |                        FORMAT   11 FORMAT(1H/,10X,8HMAXIMUM=
                4HP.M.)                                     ,F8.1,2X,6HC.F.S.,5X,4HTIME,
                         |                                  3X,F5.2,2X,
                      --------
                      |206   |
                      --------

205             CONTINUE
                                                  FORMAT   12 FORMAT(1H/,10X,8HMAXIMUM=
                4HA.M.),                         ,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.?,2X,
```

ORIGINAL PAGE IS
OF POOR QUALITY

206

```
        IF               T
   CONOPT(7          *++++   WRITE(6,13)DATE,
*) .EQ. 1 .AND. SDEP*
*TH .GT. 0.0
        F
SDEPTH,STMD,SAX,TANSM,SPLW
                                    FORMAT    13 FORMAT(3X,I4,2X,7HSDEPTH=
  2X,6HTANSM=,F6.2,2X,5HSPLW=,F6.2) ,F8.2,2X,5HSTMD=,F6.2,2X,4HSAX=,F6.2,

| MAXI =MAXI+1 |

C  MONTHLY SUMMARY STORAGE

STMROS(MAXI,1)=DSSF(DAY)

| TMSTF(MAXI,1) =AMSTF
| AMSTF=0.0
| TMRTF(MONTH) =AMRTF
| AMRTF=0.0

EMBFNX(MAXI,1)=BFNX

TMPREC(MAXI,1)=AMPREC

| AMPREC =0.0
| TMRPM(MONTH) =AMRPM
| AMRPM=0.0
| TMBF(MAXI,1) =AMBF
| AMBF =0.0
| TMIF(MAXI,1) =AMIF
| AMIF =0.0
| TMSE(MAXI,1) =AMSE
| AMSE =0.0
| TMPET(MONTH) =AMPET
| AMPET=0.0
| TMNET(MONTH) =AMNET
| AMNET=0.0
| TMSNE(MONTH) =AMSNE
| AMSNE=0.0

TMFSIL(MONTH) = AMFSIL

| AMFSIL =0.0
| EMGWS(MAXI,1) =GWS
| UZC  =SUZC*AEX90+BUZC*EXP(-2.7
|         *LZS/LZC)

        IF               T
   UZC .LT.          *++++   UZC = 0.25
* 0.25
        F

| EMUZC(MAXI,1) =UZC
| EMUZS(MAXI,1) =UZS

EMSIAM(MAXI,1)=SIAM

| EMLZS(MAXI,1) =LZS
| EMIFS(MAXI,1) =IFS
```

```
220         CONTINUE
                       |
                       *
                  . *     * .
            . *      IF        * .   T    ----------------------------------
          *     MDAY.EQ.           *++++ |        MDAY=59                 |
          *337                     *     |                                |
            * .                 . *       ----------------------------------
                  * .     . *
                  F  |+++++++++++++++++++++++++++++++++++++++++++++++++++++++++
      C
      C STORE MAXIMUM DAILY STREAM FLOW FOR YEAR
                       |
            -------------------------------------------
            | TFMAXY(MAXI) =TFMAX                      |
            -------------------------------------------
                       |
3001        CONTINUE
                       |
                       *
                  . *     * .
            . *      IF        * .   T    ----------------------------------
          *     DAY.EQ.E           *++++ |        GO TO 221               |
          *DATE                    *     |                                |
            * .                 . *       ----------------------------------
                  * .     . *
                  F  |
            CALL DAYNXT(DAY,DPY)
                       |
                       *
                  . *     * .
            . *      IF        * .   T    ----------------------------------
          *     DAY.GT.L           *++++ |        MDAY=LDAY               |
          *DAY                     *     |                                |
            * .                 . *       ----------------------------------
                  * .     . *
                  F  |+++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                     |
                       *
                  . *     * .
            . *      IF        * .   T    ----------------------------------
          *     DAY.EQ.3           *++++ |        MDAY=337                |
          *66                      *     |                                |
            * .                 . *       ----------------------------------
                  * .     . *
                  F  |+++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                  --------
                  |152   |
                  --------

      C  END OF DAY LOOP

221         CONTINUE
            RETURN

END
```

```
SUBROUTINE  ZEROP(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY)

      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
      KFLAG,IDFLAG,
      IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
        NCTRI,CTRI,FIRR,RICY,DPSE,BDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,
        XDNFS,FFOK,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
       SATRI,UHFA,
      MNRD,
        FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
        BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,
        BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
        IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPNET,FPER,
        TPLR,VINTCR,HSE,MRTRI,SPIF,CBF,SPDR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
        RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
        WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
        SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
      DEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
      TOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
      TSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
      TCRARY(1,6,1),
      DRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)

      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
      TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
      EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
      TMSTFI,T20OFH,T20PRH,TMRTFI,JULDAT,
      TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSUP,RGPM,NDSDR,YR1,
      TRHF,
      SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS

C  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)

C     BASED ON STANFORD WATERSHED MODELS III & IV

      DIMENSION BTRI(99),CONOPT(16),CRFMI(22),CTRI(99),DDIW(366),
        DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
        DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
       EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
       EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
        RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),
       THSF(24),TITLE(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
       TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
       TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T20OFH(21),T20PRH(21),
       UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
       PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)

      LOGICAL  LSHFT

      INTEGER  CDSDR,CN,CONOPT,DATE,DAY,DPY,EHSGD,HOUR,HRF,HRL,PDAY,
        PRD,RHPD,RHPH,RSBC,SGMC,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX

      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY

      INTEGER  DATES,EDATE

      DIMENSION RPLOTC(1832),RCOMMA(12087)

      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))

      REAL  IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT

      DATA  MEDCY/  0,         31,59,90,120,151,181,212,243,273,304,334/

      DATA  MEDWY/304,334,365,31,59,90,120,151,181,212,243,273        /

      REAL      MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD

      REAL  SSQM,SSQDI,SSQMI,VDRSF,VDSSF

      REAL  VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
            SMSQM

      REAL  MDRSF,MDSSF,MMRSF,MMSSF

      BACKSPACE 11

       READ  (11)
       (RPLOTC(I),I=1,1832)
```

```
      BACKSPACE 18

      READ(18)
      (RCOMMA(I),I=1,12087)

150   CONTINUE

      DATE =PDATE
      DAY  =DATES
      MDAY =PDAY
      SINDEX =0
      MAXI =0
      AMRPM=0.0
      AMPREC =0.0
      AMBF =0.0
      AMSE =0.0
      AMSTF=0.0
      AMRTF=0.0

      IF DPY .EQ. 366      T    MEDWY(5)=366
                           F

                           FORMAT   3 FORMAT(1H1,1X,'NO PRECIP
                                    HOURLY CFS VALUES',//,1X,4A4)
      WRITE(6,3)
      MSBDIC

C  BEGIN DAY LOOP

      IMONTH =MONTH

152   TDSF =0.0

      IF MONTH.NE .4       T    GO TO 148
                           F

      IF MDAY.NE. 31       T    GO TO 149
                           F

148   IF DATE.GT.
      (MOD(DAY,MDAY))      T    IMONTH=MONTH+1
                           F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
149         CONTINUE
               |
      ------------------------------------
      | PET   =EPCM(IMONTH)*DPET(DAY)    |
      | PETU  =PET                       |
      | TFMAX=0.0                        |
      ------------------------------------
               |
    C  EVAPOTRANSPIRATION ADJUSTMENTS
               |
                                        ---------------------
         IF                    T        |  GO TO 153        |
     CONOPT(7               *++++       |                   |
  *) .NE. 1                             ---------------------
         F
               |
                                        ---------------------
         IF                    T        |  PET = 0.0        |
     DMXT(DAY               *++++       |                   |
  *) - 4.0*ELDIF .LT.                   |                   |
  *40.0                                 ---------------------
         F  |+++++++++++++++++++++++++++++++
               |
                                        ---------------------
         IF                    T        |  PET = FFOR*PET   |
     SPTW .GT               *++++       |                   |
  *. SPTWCC                             ---------------------
         F  |+++++++++++++++++++++++++++++++

    C  CALCULATION OF SNOW EVAPORATION
               |
                                        ---------------------
         IF                    T        |  GO TO 153-       |
     DMNT(DAY               *++++       |                   |
  *) .GT. 32.0 .OR. SP*                 |                   |
  *TW .LE. DPSE(DAY)                    ---------------------
         F
               |
      ------------------------------------
      | SE    =DPSE(DAY)                 |
      | AMSNE=AMSNE+SE                   |
      | SPTW  =SPTW-SE                   |
      ------------------------------------
               |
                                        ---------------------------
         IF                    T        |  SDEPTH = SDEPTH - SE/SF|
     SEMD .GT               *++++       | MD                      |
  *. 0.0                                ---------------------------
         F  |+++++++++++++++++++++++++++++++

              +-------------+
153           + DO LOOP TO  +
    ++++++++++  STMT #  202 +
    +         + HOUR = 1,24 +
    +         +-------------+
    +                |
154 +                                   ---------------------------
    +    IF                    T        |  RGPM = DRGPM(DAY)      |
    +  HOUR .EQ             *++++       |                         |
    + *. SGRT + 1                       ---------------------------
    +    F  |+++++++++++++++++++++++++++++++
```

```
                 IF                  T   HSE = (FWTR*PET)/12.0
           HOUR  .EQ             *++++
        *. 9
              F

                 IF                  T   HSE = 0.0
           HOUR  .EQ             *++++
        *.21
              F

      PRH    =0
      AMPREC =AMPREC+PRH

C  ENTER SNOWMELT SUBROUTINE

                 IF                  T   CALL SNOMEL(BDDFSM,SPTW
           CONOPT(7              *++++   CC,SPM,ELDIF,DAY,
        *) .EQ. 1
              F
      SPBFLW, XDNFS,FFOR,FFSI,MRNSM,DSMGH,SDEPTH,STMD, PXCSA,HOUR,
      SAX,SOFRF,OFRFIS,SOFRFI,AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,
      WT4AM,WT4PM,ASM,ASMRG,SASFX,SARAX,DMXT,DMNT,RICY,FIRR)

155   AMRPM=AMRPM+PRH

156   TOFR =0.0
      ARHF =0.0

C  15 MINUTE ACCOUNTING AND ROUTING LOOP

      DO LOOP TO
      STMT #   187
      PRD = 1,4

      PEBI =0.0
      PPI  =0.0
      OFR  =0.0
      OFRIS=0.0
      WI   =0.0
      WEIFS=0.0
      PMEUZS =0.0
      PMELZS =0.0
      PMEIFS =0.0
      PMEOFS =0.0
      PEP    =0.25*PRH

                 IF                  T   CALL PREPRD(RGPM,DRHP,D
           CONOPT(2              *++++   AY,HOUR,DPY,PRD,PEP,
        *) .EQ. 1
              F
```

```
PRH)

IF PEP .GT. C.0          T   GO TO 157
F

IF OFUS .GT 0.0          T   GO TO 159
F

IF IFS .GT. 0.0          T   GO TO 170
F

IF NRTRI .GT. 0          T   GO TO 172
F

TRHF =0.0

IF RHFO .GT 0.0          T   GO TO 181
F

184

C   RAINFALL UPPER ZONE INTERACTION

157
IF PEP .GE. VINTCR       T   GO TO 158
F

UZS    =UZS+PEP*TPLR
VINTCR =VINTCR-PEP
PPI    =0.0
PEBI   =0.0
PMEUZS =PEP
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
IF OFUS .GT. 0.0   T → GO TO 159
F → 170

158  PPI   =PEP-VINTCR
     UZS   =UZS+VINTCR*TPLR
     VINTCR =0.0
     LZSR  =LZS/LZC
     UZC   =SUZC*AEX90+BUZC*EXP(-2.7
            *LZSR)

IF UZC .LT. 0.25   T → UZC = 0.25
F

     UZRX =2.0*ABS(UZS/UZC-1.0)+1.0
     FMR  =(1.0/(1.0+UZRX))**UZRX

IF UZS .GT. UZC   T → FMR = 1.0 - FMR
F

     PEBI =PPI*FMR
     PMEUZS =PEP-PEBI
     UZS  =UZS+PPI-PEBI

C  LOWER ZONE AND GROUNDWATER INFILTRATION

159  LZSR =LZS/LZC
     EID  =4.0*LZSR

IF LZSR .LE. 1.0   T → GO TO 160
F

     EID  =4.0+2.0*(LZSR-1.0)

IF LZSR .LE. 2.0   T → GO TO 160
F
```

```
160      EID  =6.0

         PEBI =PEBI+OFUS
         CMIR =0.25*SIAM*BMIR/(2.0**EID)
         CIVM =BIVF*2.0**LZSR

         IF CIVM .LT. 1.0           T  ---->  CIVM = 1.0
         F

         PEAI =PEBI*PEBI/(2.0*CMIR*CIVM)
         WI   =PEBI*PEBI/(2.0*CMIR)

         IF PEBI .GE. CMIR          T  ---->  WI = PEBI - 0.5*CMIR
         F

         IF PEBI .GE. CMIR*CIVM     T  ---->  PEAI = PEBI - 0.5*CMIR*
                                              CIVM
         F

         WEIFS=WI-PEAI

         IF PEBI .LE. OFUS          T  ---->  GO TO 161
         F

         PMELZS =(PEBI-WI)*((PEBI-OFUS)
                 /PEBI)
         PMEIFS =WEIFS*((PEBI-OFUS)/PEBI)
         PMEOFS =PEAI*((PEBI-OFUS)/PEBI)

161      CONTINUE

         IF (PEAI - OFUS) .GT. 0.0  T  ---->  GO TO 162
         F

         EQD  =(OFUS+PEAI)/2.0

         |163 |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
162  | EQD   =EQDF*((PEAI-OFUS)**0.6)                |

163
        IF                     T
     (OFUS +                  *****  | EQD = 0.5*(OFUS + PEAI)  |
 *PEAI) .GT. (2.0*EQD*
 *)
          F
          |++++++++++++++++++++++++++++++

        IF                     T
     (OFUS +                  *****  | GO TO 164  |
 *PEAI) .LE. 0.001
          F

     | OFR   =0.25*OFRF*(((OFUS+PEAI)*0.5)
     |        **1.67)*((1.0+0.6*((OFUS
     |         +
     |          PEAI)/(2.0*EQD))**3.0)**1.67)

        IF                     T
     OFR .GT.                 *****  | OFR = 0.75*PEAI  |
 * (0.75*PEAI)
          F
          |++++++++++++++++++++++++++++++

164
        IF                     T
     FIMP .EQ                 *****  | GO TO 168  |
 *. 0.0
          F

165  | PEIS =PPI+OFUSIS                              |

        IF                     T
     (PEIS -                  *****  | GO TO 166  |
 *OFUSIS) .GT. 0.0
          F

     | EQDIS=(OFUSIS+PEIS)/2.0                       |

          |167 |

166  | EQDIS=EQDFIS*((PEIS-OFUSIS)**0.6)             |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
167   IF (OFUSIS + PEIS) .GT. (2.0*EQDIS)   T ---> EQDIS = 0.5*(OFUSIS + PEIS)
      F
      IF (OFUSIS + PEIS) .LE. 0.01   T ---> GO TO 168
      F
      OFRIS=0.25*OFRFIS*(((OFUSIS+PEIS)
            *0.5)**1.67)*((1.0+0.6*((
               OFUSIS + PEIS)/(2.0*EQDFIS))**3.0)**1.67)
      IF OFRIS .GT. PEIS   T ---> OFRIS = PEIS
      F
168   TOFR =TOFR+FPER*OFR+FIMP*OFRIS
            +PPI*FWTR
      OFUSIS =PEIS-OFRIS
      OFUS =PEAI-OFR
      IF OFUS .GE. 0.001   T ---> GO TO 169
      F
      LZS  =LZS+OFUS
      OFUS =0.0
      OFRIS=OFRIS+OFUSIS
      OFUSIS =0.0
169   LZRX =1.5*ABS(LZS/LZC-1.0)+1.0
      FMR  =(1.0/(1.0+LZRX))**LZRX
      IF LZS .LT. LZC   T ---> FMR = 1.0 - FMR*(LZS/LZC)
      F
      PLZS =FMR*(PEBI-WI)
      PGW  =(1.0-FMR)*(PEBI-WI)*(1.0
            -SUBWF)*FPER
      GWS  =GWS+PGW
      BFNX =BFNX+PGW
      LZS  =LZS+PLZS
      IFS  =IFS+WEIFS*FPER
```

```
170 +      -------------------------------------------
    +     | SPIF =IFRL*IFS                            |
    +     | AMIF =AMIF+SPIF                           |
    +     | IFS  =IFS-SPIF                            |
    +      -------------------------------------------
    +                       |
    +                       .               -------------------------
    +               . * * * * .            |                         |
    +          . *      IF       * .   T   |    GO TO 171            |
    +       *      IFS .GE.           *++++|                         |
    +       * C.C001                  *    |                         |
    +          * .               . *        -------------------------
    +               * . * *
    +                 F |
    +      -------------------------------------------
    +     | LZS  =LZS+IFS                             |
    +     | IFS  =0.0                                 |
    +      -------------------------------------------
    +                       |
171 +      -------------------------------------------
    +     | UHFA(1) =FPER*CFR+PPI*FWTR+FIMP           |
    +     |          *OFRIS+SPIF                      |
    +     | SPDR =UHFA(1)                             |
    +      -------------------------------------------
    +                       |
    + C   ROUTING
    +                       |
172 +                       .               -------------------------
    +               . * * * * .            |                         |
    +          . *      IF       * .   T   |    GO TO 173            |
    +       *     CONOPT(1            *++++|                         |
    +       *2) .NE. 1                *    |                         |
    +          * .               . *        -------------------------
    +               * . * *
    +                 F |
    +      -------------------------------------------
    +     | URHF =URHF+0.25*UHFA(1)                   |
    +      -------------------------------------------
    +                       |
    +                       .               -------------------------
    +               . * * * * .            |                         |
    +          . *      IF       * .   T   |    GO TO 181            |
    +       *      PRD .NE.           *++++|                         |
    +       * 4                       *    |                         |
    +          * .               . *        -------------------------
    +               * . * *
    +                 F |
    +      -------------------------------------------
    +     | UHFA(1) =URHF                             |
    +      -------------------------------------------
    +                       |
173 +      -------------------------------------------
    +     | TRHF =0.0                                 |
    +     | KTRI =NCTRI                               |
    +      -------------------------------------------
    +                       |
    +                       .               -------------------------
    +               . * * * * .            |                         |
    +          . *      IF       * .   T   |    KTRI = NCSTRI        |
    +       *     CONOPT(1            *++++|                         |
    +       *3) .EQ. 1                *    |                         |
    +          * .               . *        -------------------------
    +               * . * *
    +                 F |                             +
    +                   |++++++++++++++++++++++++++++++
174 +      -------------------------------------------
    +     | URHF =UHFA(KTRI)                          |
    +      -------------------------------------------
    +                       |
    +                       .               -------------------------
    +               . * * * * .            |                         |
    +          . *      IF       * .   T   |    GO TO 176            |
    +       *     URHF .LE            *++++|                         |
    +       *. 0.0                    *    |                         |
    +          * .               . *        -------------------------
    +               * . * *
    +                 F |
```

```
175   | TRHF =TRHF+URHF*CTRI(KTRI)                |

              IF                       T
         CONOPT(1               *++++    TRHF = TRHF +
      *3) .EQ. 1 .AND. LSH*
      *FT .AND. KTRI .GE. *
              F
   URHF*SATRI(KTRI - 1)
  UHFA(KTRI + 1) = URHF

        |177  |

176   UHFA(KTRI+ 1) = 0.0

177   | KTRI =KTRI-1                               |

              IF                       T
          KTRI .GE.             *++++    GO TO 174
      *. 1
              F

178           IF                       T
         URHF .LE.              *++++    GO TO 179
      *. 0.0
              F

      | NRTRI=NCTRI                                |

              IF                       T
         CONOPT(1               *++++    NRTRI = MXTRI
      *3) .EQ. 1
              F

179   | NRTRI=NRTRI-1                              |
      | UHFA(1) =0.0                               |

              IF                       T
         CONOPT(1               *++++    GO TO 180
      *3) .NE. 1
              F

      | NNSTRI =NCSTRI+1                           |
      | UHFA(NNSTRI) =0.0                          |
```

```
      +
180   +     ------------------------------------------------------
      +     | URHF =0.0                                          |
      +     ------------------------------------------------------
      +                  |
181   +                  |                          ------------------------------------------
      +               * . * .                      |                                          |
      +         . *    IF     *  .    T            |    SRX = CSRX                            |
      +       *    SRX  .LE.          *++++        |                                          |
      +       * CSRX                  *            |                                          |
      +          * .           . *                  ------------------------------------------
      +             *  .  *                                              +
      +            F   |++++++++++++++++++++++++++++++++++++++++++++++++++
      +     ------------------------------------------------------
      +     | RHF1 =TRHF-SRX*(TRHF-RHFO)                         |
      +     | RHFO =RHF1                                         |
      +     ------------------------------------------------------
      +                  |
      +                  |                          ------------------------------------------
      +               * . * .                      |                                          |
      +         . *    IF     *  .    T            |    RHFO = 0.0                            |
      +       *    RHFO  .LT          *++++        |                                          |
      +       *. RHFMC                *            |                                          |
      +          * .           . *                  ------------------------------------------
      +             *  .  *                                              +
      +            F   |++++++++++++++++++++++++++++++++++++++++++++++++++
      +     ------------------------------------------------------
      +     | TFCFS=(4.0*RHF1+CBF-HSE)*WCFS                      |
      +     ------------------------------------------------------
      +                  |
      +                  |                          ------------------------------------------
      +               * . * .                      |                                          |
      +         . *    IF     *  .    T            |    GO TO 182                             |
      +       *    CONOPT(1               *++++    |                                          |
      +       *3)  .NE. 1                 *        |                                          |
      +          * .           . *                  ------------------------------------------
      +             *  .  *
      +            F   |
      +                |                            ------------------------------------------
      +               * . * .                      |                                          |
      +         . *    IF     *  .    T            |    GO TO 182                             |
      +       *    CONOPT(1               *++++    |                                          |
      +       *2)  .EQ. 1 .AND. PRD*               |                                          |
      +       * .NE. 4                    *        |                                          |
      +          * .           . *                  ------------------------------------------
      +             *  .  *
      +            F   |
      +
      +     CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
      +       TFCFS)
      +                  |
      +     ------------------------------------------------------
      +     | DATE =MOD(DAY,MDAY)                                |
      +     ------------------------------------------------------
      +                  |
      +                  |                          ------------------------------------------
      +               * . * .                      |                                          |
      +         . *    IF     *  .    T            |    WRITE(6,6) DATE,HOUR,PR               |
      +       *    LSHFT               *++++       |  D,NCSTRI                                |
      +          * .           . *                  ------------------------------------------
      +             *  .  *                                              +
      +            F   |++++++++++++++++++++++++++++++++++++++++++++++++++
      +                                        FORMAT     6 FORMAT(2X,I2,2X,I2,2X,I2,
      +       8HELEMENTS)                       2X,20HHISTOGRAM CHANGES TO,1X,I2,1X,
182   +     CONTINUE
      +                  |
      +                  |                          ------------------------------------------
      +               * . * .                      |                                          |
      +         . *    IF     *  .    T            |    SRX = CSRX                            |
      +       *    TFCFS  .L              *++++    |                                          |
      +       *E. 0.5*CHCAP               *        |                                          |
      +          * .           . *                  ------------------------------------------
      +             *  .  *                                              +
      +            F   |++++++++++++++++++++++++++++++++++++++++++++++++++
```

```
                       IF         T    SRX = CSRX
        (TFCFS .
   *GT. 0.5*CHCAP) .AND*
   *. (TFCFS .LT. 2.0*C*
                    F
     +(FSRX - CSRX)*((TFCFS - 0.5*CHCAP)/(1.5*CHCAP))**3

                       IF         T    SRX = FSRX
        TFCFS .G
   *T. 2.0*CHCAP
                    F

                       IF         T    GO TO 183
        TFCFS .L
   *E. TFMAX
                    F

     PRDF =PRD
     TDFP24 =HOUR

                       IF         T    TDFP24 = (TDFP24 - 1.0)
        PRD .LE.                        + 0.15*PRDF
   * 3
                    F

     TFMAX=TFCFS

183  ARHF =ARHF+RHF1

   C  STCRM OUTPUT REQUESTED BY CONOPT(1)

184
                       IF         T    GO TO 186
        CONOPT(1
   *) .NE. 1
                    F

                       IF         T    GO TO 186
        DAY .NE.
   * COSDR
                    F

                       IF         T    WRITE(6,7)
        HOUR .EQ
   *. 1 .AND. PRD .EQ.
   *1
                    F
```

```
                                        FORMAT    7 FORMAT(1H//,21X,19HRAINFALL
                                        DEPOSITION,12X,16HMOISTURE STORAGE,
    14X,17HSTREAMFLOW ORIGIN,6X,14HSTREAM OUTFLOW/2X,116HDY HR PD RA
   IN    EUZS    ELZS    EIFS    EOFS         UZS     LZS     IFS     OFS     S
   PGF    SPIF    SPBF    SPTF   INCHES       CFS)

   | DATE =MOD(DAY,MDAY)                        |

   | OFS  =OFUS*FPER+OFUSIS*FIMP
   | SPCF =OFK*FPER+OFRIS*FIMP+PPI*FWTR
   | SPBF =0.25*(CBF-HSE)
   | SPTF =SPDR+SPBF
   | SPDR =0.0

        IF                 T
     RHFO .LE          *++++|  S  TFCFS = (CBF - HSE)*WCF
   *. 0.0
        F

   | RSPTF=0.25*TFCFS/WCFS                      |

     WRITE(6,8)
     DATE,HOUR,PRD,PEP,PMEUZS,PMELZS,PMEIFS,PMEOFS,U+
   +ZS,LZS

   ,IFS,OFS,SPOF,SPIF,SPBF,SPTF,RSPTF,TFCFS
                                        FORMAT    8 FORMAT(2X,I2,1X,I2,1X,
     F7.1)                  I1,5(1X,F6.4),2X,4(F7.4),2X,5(1X,F6.4),1X,

        IF                 T
     HOUR .EQ          *++++|     GO TO 185
   *. 24 .AND. PRD .EQ.*
   * 4
        F

      |186 |

185 | NDSDP=NDSDP+1                             |

        IF                 T
     NDSDR .E          *++++|     GO TO 186
   *Q. NDSDP
        F

   CALL DAYNXT(CDSDR,DPY)
186 CONTINUE

        IF                 T
     VINTCR .          *++++|     VINTCR = VINTCR + DPET(
   *LT. 0.25*VINTMR         |    DAY)/96.0
        F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
187 ++++++++CCNTINUE
    C  END OF 15 MINUTE LOOP

              IF                  T
        CONOPT(5                 *++++  GO TO 197
    *) .NE. 1
              F

    C  HOURLY OVERLAND FLOW AND RAINFALL SORTING

              IF                  T
        TOFR  .LE                *++++  GO TO 193
    *. 0.0
              F

    | KT20 =20 |

188           IF                  T
        KT20  .LT                *++++  GO TO 192
    *. 1
              F

              IF                  T
        TOFR  .GT                *++++  GO TO 189
    *. T20OFH(KT20)
              F

        |190 |

189   T20OFH(KT20+1) = T20OFH(KT20)

        |191 |

190   T20OFH(KT20+1) = TOFR

        |193 |

191   | KT20 =KT20-1 |

        |188 |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
192   | T20DFH(1) =TOFR |

193   IF PRH .LE. 0.0   --T-->  | GO TO 197 |
        F
      | KT20 =20 |

194   IF KT20 .LT. 1    --T-->  | GO TO 196 |
        F
      IF PRH .GT. T20PRH(KT20)  --T-->  | GO TO 195 |
        F
      T20PRH(KT20 + 1) = PRH
      |197 |

195   T20PRH(KT20+1) = T20PRH(KT20)
      | KT20 =KT20-1 |
      |194 |

196   | T20PRH(1) =PRH |

C     ADDING GROUNDWATER FLCW

197   | CBF  =GWS*BFRL*(1.0+BFNRL*BFNX) |
      | GWS  =GWS-CBF                  |
      | AMBF =AMBF+CBF                 |
      | THGR =ARHF+CBF                 |

      IF HSE .GT. THGR  --T-->  | HSE = THGR |
        F
```

```
      AMSE =AMSE+HSE
      THSF(HOUR) =(THGR-HSE)*WCFS
      TDSF =TDSF+THSF(HOUR)

C
C STORE SIMULATED HOURLY STREAM FLOWS
C  DRAINING OF UPPER ZONE STORAGE

      SINDEX =SINDEX+1

      THSFD(SINDEX,2)=THSF(HOUR)

306   UZINFX =(LZS/UZC)-(LZS/LZC)

      IF UZINFX .LE. 0.0     T   GO TO 198
      F

      LZSR =LZS/LZC
      UZINLZ =0.003*BMIR*UZC*UZINFX*
             *3.0

      IF UZINLZ .GT. UZS     T   UZINLZ = UZS
      F

      UZS  =UZS-UZINLZ
      LZRX =1.5*ABS(LZSR-1.0)+1.0
      FMR  =(1.0/(1.0+LZRX))**LZRX

      IF LZS .LT. LZC        T   FMR = 1.0 - FMR*LZSR
      F

      PGW  =(1.0-FMR)*UZINLZ*(1.0-SUBWF)
           *FPER
      PLZS =FMR*UZINLZ
      LZS  =LZS+PLZS
      GWS  =GWS+PGW
      BFNX =BFNX+PGW

C  4 PM ADJUSTMENTS OF VARIOUS VALUES
```

```
198
      IF HOUR .NE. 16          T     GO TO 202
      F
      AEX90=0.9*(AEX90+PET)
      AEX96=0.96*(AEX96+PET)

C   INFILTRATION CORRECTION

      SIAM =(AEX96/AETX)**SIAC

      IF SIAM .LT. 0.33        T     SIAM = 0.33
      F
      BFNX =0.97*BFNX

      IF PET .EQ. 0.0          T     GO TO 202
      F
C   EVAP-TRANS LOSS FROM GROUNDWATER

      GWET =GWS*GWETF*PET*FPER
      GWS  =GWS-GWET
      BFNX =BFNX-GWET

      IF BFNX .LT. 0.0         T     BFNX = 0.0
      F
      AMPET=AMPET+PET

      IF PET .GE. UZS          T     GO TO 199
      F
      UZS  =UZS-PET
      AMNET=AMNET+PET
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                     |
                  --------
                  |202   |
                  --------

                     |
          ---------------------------------
199 +     | PET  =PET-UZS                 |
    +     | AMNET=AMNET+UZS               |
    +     | UZS  =0.0                     |
    +     | LZSR =LZS/LZC                 |
          ---------------------------------
                     |
            . * * * .                    ------------------------
          *     IF      * . T            |    GO TO 200         |
    +    *   PET .GE.      *++++         |                      |
    +    * ETLF*LZSR       *             ------------------------
            * . * * . *
                F |
          ---------------------------------------
    +     | SET  =PET*(1.0-PET/(2.0*ETLF*LZSR)) |
          ---------------------------------------
                     |
                  --------
                  |201   |
                  --------

                     |
          ---------------------------------
200 +     | SET  =0.5*ETLF*LZSR           |
          ---------------------------------
                     |
          ---------------------------------
201 +     | LZS  =LZS-SET                 |
    +     | AMNET=AMNET+SET               |
          ---------------------------------
                     |
202 ++++++++CCNTINUE
     C  END OF HOUR LOOP
                     |
          ---------------------------------
          | DSSF(DAY) =TDSF/24.0          |
          ---------------------------------
                     |
            . * * * .                    ------------------------
          *     IF      * . T            | DSSF(DAY) = DSSF(DAY) + |
         *   CONOPT(1      *++++         | DDIW(DAY)            |
         *1) .EQ. 1        *             ------------------------
            * . * * . *
                F |++++++++++++++++++++++++++++++++
          ---------------------------------
203       | AMRTF=AMRTF+DRSF(DAY)         |
          | AMSTF=AMSTF+DSSF(DAY)         |
          ---------------------------------
                     |
            . * * * .                    ------------------------
          *     IF      * . T            |  EDLZS(DAY) = LZS    |
         *   CONOPT(6      *++++         |                      |
         *) .EQ. 1         *             ------------------------
            * . * * . *
                F |++++++++++++++++++++++++++++++++
     C  STORE ERRORS AND FLOW DURATION
```

```
                    !
                 . * * .           -------------------------------
             . *   IF    * .   T   |                             |
           *   CONOPT(4       *++++|      GO TO 204              |
           *) .NE. 1          *    |                             |
             * .         . *       -------------------------------
                 * . *
                 F |
                   |
  --------------------------------------
  | ERR   =DSSF(DAY)-DRSF(DAY)          |
  --------------------------------------
                   |
                 . * * .           -------------------------------
             . *   IF    * .   T   |                             |
           *   DRSF(DAY       *++++|      KRFMI  =  1.0          |
           *) .LT. 1.0        *    |                             |
             * .         . *       -------------------------------
                 * . *                                   +
                 F |+++++++++++++++++++++++++++++++++++++
                   |
                 . * * .           -------------------------------
             . *   IF    * .   T   |                             |
           *   DRSF(DAY       *++++|      KRFMI=  2.0*ALOG(DRSF(DA
           *) .GT. 1.0        *    |  Y)) + 2.0                  |
             * .         . *       -------------------------------
                 * . *                                   +
                 F |+++++++++++++++++++++++++++++++++++++
                   |
  --------------------------------------
  | CRFMI(KRFMI)  =CRFMI(KRFMI)+1.0     |
  | SERR(KRFMI)  =SERR(KRFMI)+ERR       |
  | SERA(KRFMI)  =SERA(KRFMI)+ABS(ERR)  |
  | SQER(KRFMI)  =SQER(KRFMI)+ERR*ERR   |
  | SESF(KRFMI)  =0.0                   |
  --------------------------------------
                   |
                 . * * .           -------------------------------
             . *   IF    * .   T   |                             |
           *   CRFMI(KR       *++++|      SESF(KRFMI)  =  SQRT(ABS(
           *FMI) .GT. 1.0     *    |  (SQER(KRFMI)  -            |
             * .         . *       -------------------------------
                 * . *                                   +
                 F |+++++++++++++++++++++++++++++++++++++
   SERR(KRFMI)**2/CRFMI(KRFMI))/(CRFMI(KRFMI)  -  1.0)))
 204      CONTINUE
                   |
  --------------------------------------
  | DATE =DAY                           |
  --------------------------------------
                   |
                 . * * .           -------------------------------
             . *   IF    * .   T   |                             |
           *   (MONTH.E       *++++|      DATE=MOD(DAY,MDAY)     |
           *Q.4).AND.(MDAY.EQ.3*   |                             |
           *1)                *    |                             |
             * .         . *       -------------------------------
                 * . *                                   +
                 F |+++++:+++++++++++++++++++++++++++++++
                   |
                 . * * .           -------------------------------
             . *   IF    * .   T   |                             |
           *   MONTH.NE       *++++|      DATE=MOD(DAY,MDAY)     |
           *.4                *    |                             |
             * .         . *       -------------------------------
                 * . *                                   +
                 F |+++++++++++++++++++++++++++++++++++++
                   |
      +---------------------------------------------------+
     +  WRITE(6,9)                                       +
    +   DATE, (THSF(HCUR),HOUR=1,12)                    +
   +                                                   +
  +---------------------------------------------------+
                   |
```

```
                                              FORMAT      9 FORMAT(1H/,1X/,1X,I4,2X,
                                                          2HAM,1X,6F8.1,3X,6F8.1)
      WRITE(6,10)
      (THSF(HCUR),HCUR=13,24), DSSF(DAY)

                                              FORMAT     10 FORMAT(1HJ,6X,2HPM,1X,
                                                          6F8.1,3X,7F8.1)
           IF              T
        TDFP24        ++++      GO TO 205
      LT. 12.0
           F

      | TDFP12 =TDFP24-12.0                 |

                                              FORMAT     11 FORMAT(1H/,10X,8HMAXIMUM=
      4HP.M.)                              ,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,
            |206 |
205     CONTINUE
                                              FORMAT     12 FORMAT(1H/,10X,8HMAXIMUM=
      4HA.M.)                              ,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,

206
           IF              T
        CONOPT(7          ++++      WRITE(6,13)DATE,
      ) .EQ. 1 .AND. SDEP
      TH .GT. 0.0
           F
      SDEPTH,STMD,SAX,TANSM,SPLW      FORMAT     13 FORMAT(3X,I4,2X,7HSDEPTH=
                                                 ,F8.2,2X,5HSTMD=,F6.2,2X,4HSAX=,F6.2,
        2X,6HTANSM=,F6.2,2X,5HSPLW=,F6.2)

      | MAXI =MAXI+1                        |

    C  MONTHLY SUMMARY STORAGE

      STMROS(MAXI,2)=DSSF(DAY)

      | TMSTF(MAXI,2) =AMSTF
      | AMSTF=0.0
      | TMRTF(MONTH) =AMRTF
      | AMRTF=0.0

      EMBFNX(MAXI,2)=BFNX

      TMPREC(MAXI,2)=AMPREC

      | AMPREC =0.0
      | TMRPM(MONTH) =AMRPM
      | AMRPM=0.0
      | TMBF(MAXI,2) =AMBF
      | AMBF =0.0
      | TMIF(MAXI,2) =AMIF
      | AMIF =0.0
      | TMSE(MAXI,2) =AMSE
      | AMSE =0.0
      | TMPET(MONTH) =AMPET
      | AMPET=0.0
      | TMNET(MONTH) =AMNET
      | AMNET=0.0
      | TMSNE(MONTH) =AMSNE
      | AMSNE=0.0
```

```
                      |
        TMFSIL(MCNTH) = AMFSIL
                      |
        ---------------------------------------------
        | AMFSIL =0.0                               |
        | EMGWS(MAXI,2) =GWS                        |
        | UZC  =SUZC*AFX90+BUZC*EXP(-2.7            |
        |        *LZS/LZC)                          |
        ---------------------------------------------
                      |
                      .                 ------------------------------------
                .  *     *  .           |                                  |
            . *       IF       *  .  T  |     UZC = 0.25                   |
           *   UZC  .LT.          *++++ |                                  |
           * 0.25                 *     |                                  |
              *  .           .  *       ------------------------------------
                   *    .   *
                   F    |++++++++++++++++++++++++++++++++
                        |
        ---------------------------------------------
        | EMUZC(MAXI,2) =UZC                        |
        ---------------------------------------------
                      |
        ---------------------------------------------
        | EMUZS(MAXI,2) =UZS                        |
        ---------------------------------------------
                      |
        EMSIAM(MAXI,2)=SIAM
                      |
        ---------------------------------------------
        | EMLZS(MAXI,2) =LZS                        |
        | EMIFS(MAXI,2) =IFS                        |
        ---------------------------------------------
                      |
 220    CONTINUE
                      |
                      .                 ------------------------------------
                .  *     *  .           |                                  |
            . *       IF       *  .  T  |     MDAY=59                      |
           *   MDAY.EQ.           *++++ |                                  |
           *337                   *     |                                  |
              *  .           .  *       ------------------------------------
                   *    .   *
                   F    |++++++++++++++++++++++++++++++++
    C
    C STORE MAXIMUM DAILY STREAM FLOW FOR YEAR
                      |
        ---------------------------------------------
        | TFMAXY(MAXI) =TFMAX                       |
        ---------------------------------------------
                      |
 3001   CONTINUE
                      |
                      .                 ------------------------------------
                .  *     *  .           |                                  |
            . *       IF       *  .  T  |     GO TO 221                    |
           *   DAY.EQ.E           *++++ |                                  |
           *DATE                  *     |                                  |
              *  .           .  *       ------------------------------------
                   *    .   *
                   F    |
        CALL DAYNXT(DAY,DPY)
                      |
                      .                 ------------------------------------
                .  *     *  .           |                                  |
            . *       IF       *  .  T  |     MDAY=LDAY                    |
           *   DAY.GT.L           *++++ |                                  |
           *DAY                   *     |                                  |
              *  .           .  *       ------------------------------------
                   *    .   *
                   F    |++++++++++++++++++++++++++++++++
```

```
                  .
                * . *
            . *       * .
          . *    IF     * .      T    ----------------------
        *     DAY.EQ.3       *  ++++|     MDAY=337          |
       *66                     *    |                       |
          *  .             .  *      ----------------------
            *  .   .   .  *                               +
               F  |  +++++++++++++++++++++++++++++++++++++++
                  |
               ---------
               |152    |
               ---------

      C  END OF DAY LOOP
  221       CCNTINUE
            RETURN
  END
```

```
SUBROUTINE FORCST(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)

      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
      KFLAG,IDFLAG,
      IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
        NCTRI,CTRI,FIRR,RICY,DPSE,BDDFSM,SPBFLW,SPTWCC,SPM,ELDIF,
        XDNFS,FFOP,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
       SATRI,UHFA,
      MNRD,
        FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
        BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXOPV,
        BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,HFNHR,IFPRC,
        IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAET,FPER,
        TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPDR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
        RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
        WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
        SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTN,STMD,SFMD,ASMRG,
      DEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
      TCMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
      TSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
      TCRARY(1,6,1),
      DRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)

      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
      TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
      EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
      TMSTFI,T20OFH,T20PRH,TMRTFI,JULDAT,
      TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
      TRHF,
      SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS

    C KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)

    C   BASED ON STANFORD WATERSHED MODELS III & IV

      DIMENSION BTRI(99),CONOPT(16),CREMI(22),CTRI(99),DDIW(366),
        DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
        DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
       EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
       EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
        RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),
       THSF(24),TITLE(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
       TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
       TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T20OFH(21),T20PRH(21),
       UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
       PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)

      LOGICAL  LSHFT

      INTEGER  CDSDR,CN,CONOPT,DATE,DAY,DPY,EHSGD,HOUR,HRF,HRL,PDAY,
        PRD,RHPD,RHPH,RSBD,SGMD,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX

      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY

      INTEGER  DATES,EDATE,SINDET,MPDAY(15)

      DIMENSION RPLOTC(1832),RCOMMA(12087)

      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))

      REAL  IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT

      DATA  MEDCY/  0,       31,59,90,120,151,181,212,243,273,304,334/

      DATA  MEDWY/304,334,365,31,59,90,120,151,181,212,243,273         /

      REAL     MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD

      REAL  SSQM,SSQDI,SSQMI,VDRSF,VDSSF

      REAL  VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
            SMSQM

      REAL  MDRSF,MDSSF,MMRSF,MMSSF

      BACKSPACE 11

        READ  (11)
        (RPLOTC(I),I=1,1832)
```

```
        BACKSPACE 18

        READ(18)
        (RCOMMA(I),I=1,12387)

150     CONTINUE

        DATE  =PDATE
        DAY   =DATES
        MDAY  =PDAY
        SINDEX =0
        MAXI  =0
        AMRPM=0.0
        AMPREC =0.0
        AMBF =0.0
        AMSE =0.0
        AMSTF=0.0
        AMRTF=0.0

            IF
        DPY .EQ.            T     MEDWY(5)=366
        366
            F

                                  FORMAT     3 FORMAT(1H1,1X,'FORECAST
                                   CASE HOURLY CFS VALUES',//,1X,4A4)
        WRITE(6,3)
        MSBDIC

   C  BEGIN DAY LOOP

        IMONTH =MONTH

152     TDSF =0.0

            IF
        MONTH.NE.           T     GO TO 148
        4
            F

            IF
        MDAY.NE.            T     GO TO 149
        31
            F

148
            IF
        DATE.GT.            T     IMONTH=MONTH+1
        (MOD(DAY,MDAY))
            F
```

```
149          CONTINUE
                  |
        ---------------------------------------
        | PET  =EPCM(IMONTH)*DPET(DAY)        |
        | PETU =PET                           |
        | TFMAX=0.0                           |
        ---------------------------------------
                  |
C  EVAPOTRANSPIRATION ADJUSTMENTS
                  |
                  .
             . * *  * .                 ------------------------------
          . *    IF    * .  T         |     GO TO 153                |
        *    CONOPT(7          *++++  |                              |
        *) .NE. 1              *      |                              |
          * .            . *          ------------------------------
              *  .  *
               F |
                 |
                  .
             . * *  * .                 ------------------------------
          . *    IF    * .  T         |     PET = 0.0                |
        *    DMXT(DAY          *++++  |                              |
        *) - 4.0*ELDIF .LT.   *       |                              |
        *40.0                  *      |                              |
          * .            . *          ------------------------------
              *  .  *
               F |++++++++++++++++++++++++++++++
                 |
                  .
             . * *  * .                 ------------------------------
          . *    IF    * .  T         |     PET = FFOR*PET           |
        *    SPTW .GT          *++++  |                              |
        *. SPTWCC              *      |                              |
          * .            . *          ------------------------------
              *  .  *                                  +
               F |++++++++++++++++++++++++++++++
                 |
C  CALCULATION OF SNOW EVAPORATION
                  |
                  .
             . * *  * .                 ------------------------------
          . *    IF    * .  T         |     GO TO 153                |
        *    DMNT(DAY          *++++  |                              |
        *) .GT. 32.0 .OR. SP*         |                              |
        *TW .LE. DPSE(DAY)     *      |                              |
          * .            . *          ------------------------------
              *  .  *
               F |
                 |
        ---------------------------------------
        | SE   =DPSE(DAY)                     |
        | AMSNE=AMSNE+SE                      |
        | SPTW =SPTW-SE                       |
        ---------------------------------------
                  |
                  .
             . * *  * .                 ------------------------------
          . *    IF    * .  T         |     SDEPTH = SDEPTH - SE/SF  |
        *    SFMD .GT          *++++  |   MD                         |
        *. 0.0                 *      |                              |
          * .            . *          ------------------------------
              *  .  *                                  +
               F |++++++++++++++++++++++++++++++
                 |
153              +-------------+
                 +  DO LOOP TO  +
      +++++++++++  STMT #   202 +
      +          + HOUR = 1,24  +
      +          +-------------+
      +                 |
      +
      +(NSGRD IF((NSGRD .EQ. 0)  .AND. (DRHP(DAY,HOUR) .NE. 0.0)  .AND. (PET .EQ.
      +          PETU) .AND. (CONOPT(3) .EQ. 1)) PET = 0.5*PET
      +
```

```
154 +          .
      +      . * * .
      +   .     IF    *  .   T   -----------------------
      +  *   HOUR  .EQ      *++++|  RGPM = DRGPM(DAY)    |
      +  *. SGRT + 1               |                     |
      +      * .        . *        -----------------------
      +         * . *
      +         F |                          +
      +           |+++++++++++++++++++++++++++
      +
      +      . * * .
      +   .     IF    *  .   T   -----------------------
      +  *   HOUR  .EQ      *++++|  HSE = (FWTR*PET)/12.0 |
      +  *. 9                      |                     |
      +      * .        . *        -----------------------
      +         * . *
      +         F |                          +
      +           |+++++++++++++++++++++++++++
      +
      +      . * * .
      +   .     IF    *  .   T   -----------------------
      +  *   HOUR  .EQ      *++++|  HSE = 0.0            |
      +  *.21                      |                     |
      +      * .        . *        -----------------------
      +         * . *
      +         F |                          +
      +           |+++++++++++++++++++++++++++
      +
      +    ------------------------------------
      +    | PRH    =RGPM*DRHP(DAY,HOUR)       |
      +    | AMPREC =AMPREC+PRH                |
      +    ------------------------------------
      +           |
      +  C  ENTER SNOWMELT SUBROUTINE
      +           |
      +      . * * .
      +   .     IF    *  .   T   -----------------------
      +  *   CONOPT(7          *++++|   CALL SNOMEL(BDDFSM,SPTW |
      +  *) .EQ. 1                 |  CC,SPM,ELDIF,DAY,  |
      +      * .        . *        -----------------------
      +         * . *
      +         F |                          +
      +           |+++++++++++++++++++++++++++
      +     SPBFLW, XDNFS,FFOR,FFSI,MRNSM,DSMGH,SDEPTH,STMD, PXCSA,HOUR,
      +     SAX,SOFRF,OFRFIS,SOFRFI,AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,
      +     WT4AM,WT4PM,ASM,ASMRG,SASFX,SARAX,DMXT,DMNT,RICY,FIRR)
      +    ------------------------------------
155   +    | AMRPM=AMRPM+PRH                   |
      +    ------------------------------------
      +           |
      +    ------------------------------------
156   +    | TOFR =0.0                         |
      +    | ARHF =0.0                         |
      +    ------------------------------------
      +           |
      +  C  15 MINUTE ACCOUNTING AND RCUTING LOOP
      +           |
      +      +-------------+
      +      + DO LOOP TO  +
      ++++++++ STMT #   187 +
      +      + PRD = 1,4   +
      +      +-------------+
      +           |
      +    ------------------------------------
      +    | PEBI =0.0                         |
      +    | PPI  =0.0                         |
      +    | CFR  =0.0                         |
      +    | OFRIS=0.0                         |
      +    | WI    =0.0                        |
      +    | WEIFS=0.0                         |
      +    | PMEUZS =0.0                       |
      +    | PMELZS =0.0                       |
      +    | PMEIFS =0.0                       |
      +    | PMEOFS =0.0                       |
      +    | PEP   =0.25*PRH                   |
      +    ------------------------------------
      +           |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
           IF                      CALL PREPRD(RGPM,DRHP,D
       CONUPT(2               T   AY,HOUR,DPY,PRD,PEP,
   *) .EQ. 1                +++++
             F
     PRH)        ++++++++++++++++++++++++++++++

           IF              T       GO TO 157
       PEP .GT.             +++++
   * 0.0
             F

           IF              T       GO TO 159
       OFUS .GT             +++++
   *. 0.0
             F

           IF              T       GO TO 170
       IFS .GT.             +++++
   * 0.0
             F

           IF              T       GO TO 172
       NRTRI .G             +++++
   *T. 0
             F

   | TRHF =0.0                                  |

           IF              T       GO TO 181
       RHFO .GT             +++++
   *. 0.0
             F

         |184 |

   C  RAINFALL UPPER ZONE INTERACTION

157        IF              T       GO TO 158
       PEP .GE.             +++++
   * VINTCR
             F

   | UZS    =UZS+PEP*TPLR                       |
   | VINTCR =VINTCR-PEP                         |
   | PPI    =0.0                                |
   | PEBI   =0.0                                |
   | PMEUZS =PEP                                |
```

```
IF OFUS .GT. 0.0                         T -> GO TO 159
F
170

158   PPI    =PEP-VINTCR
      UZS    =UZS+VINTCR*TPLR
      VINTCR =0.0
      LZSR   =LZS/LZC
      UZC    =SUZC*AEX90+BUZC*EXP(-2.7
              *LZSR)

IF UZC .LT. 0.25                         T -> UZC = 0.25
F

      UZRX =2.0*ABS(UZS/UZC-1.0)+1.0
      FMR  =(1.0/(1.0+UZRX))**UZRX

IF UZS .GT. UZC                          T -> FMR = 1.0 - FMR
F

      PEBI   =PPI*FMR
      PMEUZS =PEP-PEBI
      UZS    =UZS+PPI-PEBI

C  LOWER ZONE AND GROUNDWATER INFILTRATION

159   LZSR =LZS/LZC
      EID  =4.0*LZSR

IF LZSR .LE. 1.0                         T -> GO TO 160
F

      EID  =4.0+2.0*(LZSR-1.0)

IF LZSR .LE. 2.0                         T -> GO TO 160
F
```

```
      EID  =6.0

160   PEBI =PEBI+OFUS
      CMIR =0.25*SIAM*BMIR/(2.0**EID)
      CIVM =BIVF*2.0**LZSR

      IF  CIVM .LT 1.0          T   CIVM = 1.0
                                F

      PEAI =PEBI*PEBI/(2.0*CMIR*CIVM)

      WI   =PEBI*PEBI/(2.0*CMIR)

      IF  PEBI .GE CMIR         T   WI = PEBI - 0.5*CMIR
                                F

      IF  PEBI .GE CMIR*CIVM    T   PEAI = PEBI - 0.5*CMIR*
                                      CIVM
                                F

      WEIFS=WI-PEAI

      IF  PEBI .LE OFUS         T   GO TO 161
                                F

      PMELZS =(PEBI-WI)*((PEBI-OFUS)
             /PEBI)
      PMEIFS =WEIFS*((PEBI-OFUS)/PEBI)
      PMEOFS =PEAI*((PEBI-OFUS)/PEBI)

161   CONTINUE

      IF  (PEAI -
      OFUS) .GT. 0.0            T   GO TO 162
                                F

      EQD  =(OFUS+PEAI)/2.0
```

```
             |
          |163  |

             |
162   | EQD  =EQDF*((PEAI-OFUS)**0.6)          |
             |
163
     IF (OFUS +                T        EQD = 0.5*(OFUS + PEAI)
     PEAI) .GT. (2.0*EQD*      *++++
     )
           F
             |++++++++++++++++++++++++++++++

     IF (OFUS +                T        GO TO 164
     PEAI) .LE. 0.001          *++++
           F
             |
      | OFR   =0.25*OFRF*(((OFUS+PEAI)*0.5)
      |          **1.67)*((1.0+0.6*((OFUS
      |          +
      |           PEAI)/(2.0*EQD))**3.0)**1.67)
             |
     IF OFR .GT.               T        OFR = 0.75*PEAI
     (0.75*PEAI)               *++++
           F
             |++++++++++++++++++++++++++++++
164
     IF FIMP .EQ               T        GO TO 168
     0.0                       *++++
           F
             |
165   | PEIS =PPI+OFUSIS                       |
             |
     IF (PEIS -                T        GO TO 166
     OFUSIS) .GT. 0.0          *++++
           F
             |
      | EQDIS=(OFUSIS+PEIS)/2.0                |
             |
          |167  |

             |
166   | EQDIS=EQDFIS*((PEIS-OFUSIS)**0.6)      |
```

```
167 +                  |
    +            * * * . * .       -------------------------------
    +        . *     IF     * .  T |   EQDIS = 0.5*(OFUSIS + P   |
    +      *     (OFUSIS        *++++|  EIS)                       |
    +      *+ PEIS) .GT. (2.0*E*    |                             |
    +      *QDIS)                *  |                             |
    +        * .            . *    -------------------------------
    +            * . *
    +            F |                                   +
    +              |++++++++++++++++++++++++++++++++++
    +              |
    +            * * * . * .       -------------------------------
    +        . *     IF     * .  T |   GO TO 168                 |
    +      *     (OFUSIS        *++++|                             |
    +      *+ PEIS) .LE. 0.01    *  |                             |
    +        * .            . *    -------------------------------
    +            * . *
    +            F |
    +              |
    +   ------------------------------------------
    +   | OFRIS=0.25*OFRFIS*(((OFUSIS+PEIS)       |
    +   |        *0.5)**1.67)*((1.0+0.6*((        |
    +   |           OFUSIS + PEIS)/(2.0*EQDFIS))**3.0)**1.67)
    +   ------------------------------------------
    +              |
    +            * * * . * .       -------------------------------
    +        . *     IF     * .  T |   OFRIS = PEIS              |
    +      *     OFRIS .G       *++++|                             |
    +      *T. PEIS              *  |                             |
    +        * .            . *    -------------------------------
    +            * . *
    +            F |                                   +
    +              |++++++++++++++++++++++++++++++++++
    +              |
168 +   ------------------------------------------
    +   | TOFR =TOFR+FPER*OFR+FIMP*OFRIS          |
    +   |        +PPI*FWTR                        |
    +   | OFUSIS =PEIS-OFRIS                      |
    +   | OFUS =PEAI-OFR                          |
    +   ------------------------------------------
    +              |
    +            * * * . * .       -------------------------------
    +        . *     IF     * .  T |   GO TO 169                 |
    +      *     OFUS .GE       *++++|                             |
    +      *. 0.001              *  |                             |
    +        * .            . *    -------------------------------
    +            * . *
    +            F |
    +              |
    +   ------------------------------------------
    +   | LZS  =LZS+OFUS                          |
    +   | OFUS =0.0                               |
    +   | OFRIS=OFRIS+OFUSIS                      |
    +   | OFUSIS =0.0                             |
    +   ------------------------------------------
    +              |
169 +   ------------------------------------------
    +   | LZRX =1.5*ABS(LZS/LZC-1.0)+1.0          |
    +   | FMR  =(1.0/(1.0+LZRX))**LZRX            |
    +   ------------------------------------------
    +              |
    +            * * * . * .       -------------------------------
    +        . *     IF     * .  T |   FMR = 1.0 - FMR*(LZS/LZ   |
    +      *     LZS .LT.       *++++|  C)                         |
    +      * LZC                 *  |                             |
    +        * .            . *    -------------------------------
    +            * . *
    +            F |                                   +
    +              |++++++++++++++++++++++++++++++++++
    +              |
    +   ------------------------------------------
    +   | PLZS =FMR*(PEBI-WI)                     |
    +   | PGW  =(1.0-FMR)*(PEBI-WI)*(1.0          |
    +   |        -SUBWF)*FPER                     |
    +   | GWS  =GWS+PGW                           |
    +   | BFNX =BFNX+PGW                          |
    +   | LZS  =LZS+PLZS                          |
    +   | IFS  =IFS+WEIFS*FPER                    |
    +   ------------------------------------------
    +              |
```

```
170   | SPIF =IFRL*IFS
      | AMIF =AMIF+SPIF
      | IFS  =IFS-SPIF

        IF IFS .GE. 0.0001          T ---> GO TO 171
        F

      | LZS  =LZS+IFS
      | IFS  =0.0

171   | UHFA(1) =FPER*OFR+PPI*FWTR+FIMP
      |         *OFRIS+SPIF
      | SPDR =UHFA(1)

C  ROUTING

172     IF CONOPT(12) .NE. 1        T ---> GO TO 173
        F

      | URHF =URHF+0.25*UHFA(1)

        IF PRD .NE. 4               T ---> GO TO 181
        F

      | UHFA(1) =URHF

173   | TRHF =0.0
      | KTRI =NCTRI

        IF CONOPT(13) .EQ. 1        T ---> KTRI = NCSTRI
        F

174   | URHF =UHFA(KTRI)

        IF URHF .LE. 0.0            T ---> GO TO 176
        F
```

```
     +                        |
     +  ---------------------------------------------------
 175 +  | TRHF =TRHF+URHF*CTRI(KTRI)                      |
     +  ---------------------------------------------------
     +                        |
     +                        .                      ------------------------------------
     +                 . *        * .               |                                   |
     +            . *        IF         * .    T    |      TRHF = TRHF +                |
     +         *      CONOPT(1             *++++    |                                   |
     +         *3)  .EQ. 1  .AND. LSH*              |                                   |
     +         *FT  .AND. KTRI  .GE.  *             |                                   |
     +            *  .            .  *              ------------------------------------
     +                 *  .  *                                                   +
     +                  F  |                                                     +
     +                     |+++++++++++++++++++++++++++++++++++++++++++++++++++++++
     +     URHF*SATRI(KTRI - 1)
     +  UHFA(KTRI + 1) = URHF
     +                     |
     +                  -------
     +                  |177  |
     +                  -------
     +
     +
 176 +  UHFA(KTRI+ 1) = 0.0
     +                     |
     +  ---------------------------------------------------
 177 +  | KTRI =KTRI-1                                    |
     +  ---------------------------------------------------
     +                     |
     +                     .                     ------------------------------------
     +                 . *    * .               |                                   |
     +            . *      IF      * .    T     |      GO TO 174                    |
     +         *     KTRI  .GE            *++++ |                                   |
     +         *.  1                      *     |                                   |
     +            *  .            .  *          ------------------------------------
     +                 *  .  *
     +                  F  |
     +                     |
 178 +                     .                     ------------------------------------
     +                 . *    * .               |                                   |
     +            . *      IF      * .    T     |      GO TO 179                    |
     +         *     URHF  .LE            *++++ |                                   |
     +         *.  0.0                    *     |                                   |
     +            *  .            .  *          ------------------------------------
     +                 *  .  *
     +                  F  |
     +                     |
     +  ---------------------------------------------------
     +  | NRTRI=NCTRI                                     |
     +  ---------------------------------------------------
     +                     |
     +                     .                     ------------------------------------
     +                 . *    * .               |                                   |
     +            . *      IF      * .    T     |      NRTRI = MXTRI                |
     +         *      CONOPT(1            *++++ |                                   |
     +         *3)  .EQ. 1                *     |                                   |
     +            *  .            .  *          ------------------------------------
     +                 *  .  *                                              +
     +                  F  |++++++++++++++++++++++++++++++++++++++++++++++++++
     +                     |
     +  ---------------------------------------------------
 179 +  | NRTRI=NRTRI-1                                   |
     +  | UHFA(1) =0.0                                    |
     +  ---------------------------------------------------
     +                     |
     +                     .                     ------------------------------------
     +                 . *    * .               |                                   |
     +            . *      IF      * .    T     |      GO TO 180                    |
     +         *      CONOPT(1            *++++ |                                   |
     +         *3)  .NE. 1                *     |                                   |
     +            *  .            .  *          ------------------------------------
     +                 *  .  *
     +                  F  |
     +                     |
     +  ---------------------------------------------------
     +  | NNSTRI =NCSTRI+1                                |
     +  | UHFA(NNSTRI) =0.0                               |
     +  ---------------------------------------------------
```

```
180   | URHF =0.0 |

181   IF SRX .LE. CSRX   --T-->  | SRX = CSRX |
      F

      | RHFI =TRHF-SRX*(TRHF-RHFO) |

      | RHFO =RHF1 |

      IF RHFO .LT RHFMC   --T-->  | RHFC = 0.0 |
      F

      | TFCFS=(4.0*RHF1+CBF-HSE)*WCFS |

      IF CONOPT(13) .NE. 1   --T-->  | GO TO 182 |
      F

      IF CONOPT(12) .EQ. 1 .AND. PROD .NE. 4   --T-->  | GO TO 182 |
      F

      CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
        TFCFS)

      | DATE =MOD(DAY,MDAY) |

      IF LSHFT   --T-->  | WRITE(6,6) DATE,HOUR,PROD,NCSTRI |
      F

                                       FORMAT     6 FORMAT(2X,I2,2X,I2,2X,
      8HELEMENTS)                  I2,2X,20HHISTOGRAM CHANGES TO,1X,I2,1X,

182   CONTINUE

      IF TFCFS .LE. 0.5*CHCAP   --T-->  | SRX = CSRX |
      F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                                                   SRX = CSRX
            IF                          T
       (TFCFS .                        *++++
*GT. 0.5*CHCAP) .AND*
*. (TFCFS .LT. 2.0*C*

       F
   +(FSRX - CSRX)*((TFCFS - 0.5*CHCAP)/(1.5*CHCAP))**3

            IF                          T          SRX = FSRX
      TFCFS .G                         *++++
*T. 2.0*CHCAP
       F

            IF                          T          GO TO 183
      TFCFS .L                         *++++
*E. TFMAX
       F

  | PRDF =PRD
    TDFP24 =HOUR

            IF                          T          TDFP24 = (TDFP24 - 1.0)
      PRD .LE.                         *++++          + 0.15*PRDF
* 3
       F

  | TFMAX=TFCFS

183  | ARHF =ARHF+RHF1

C    STORM OUTPUT REQUESTED BY CONOPT(1)

184
            IF                          T          GO TO 186
      CONOPT(1                         *++++
*) .NE. 1
       F

            IF                          T          GO TO 186
      DAY .NE.                         *++++
* CDSDR
       F

            IF                          T          WRITE(6,7)
      HOUR .EQ                         *++++
*. 1 .AND. PRD .EQ. *
*1
       F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                                                  FORMAT     7 FORMAT(1H//,21X,19HRAINFALL
                                                          DEPOSITION,12X,16HMOISTURE STORAGE,
   14X,17HSTREAMFLOW ORIGIN,6X,14HSTREAM OUTFLOW/2X,116HDY HR PD RA
  IN    EUZS    ELZS    EIFS    EOFS        UZS      LZS      IFS      OFS      S
  PCF    SPIF    SPBF    SPTF   INCHES      CFS)

  DATE =MOD(DAY,MDAY)
  CFS  =OFUS*FPER+OFUSIS*FIMP
  SPOF =OFR*FPEP+OFRIS*FIMP+PPI*FWTR
  SPBF =0.25*(CBF-HSE)
  SPTF =SPDR+SPBF
  SPDR =0.0

  IF RHFO .LE. 0.0        T  ---->  TFCFS = (CBF - HSE)*WCF
                                    S
  F

  RSPTF=0.25*TFCFS/WCFS

  WRITE(6,8)
  DATE,HOUR,PRD,PEP,PMEUZS,PMELZS,PMEIFS,PMEOFS,U+
  +ZS,LZS

  ,IFS,OFS,SPCF,SPIF,SPBF,SPTF,RSPTF,TFCFS
                                                 FORMAT      8 FORMAT(2X,I2,1X,I2,1X,I1,
   F7.1)                                        5(1X,F6.4),2X,4(F7.4),2X,5(1X,F6.4),1X,

  IF HOUR .EQ. 24 .AND. PRD .EQ. 4      T  ---->  GO TO 185
  F

  186

185  NDSCP=NDSCP+1

  IF NDSDR .EQ. NDSDP      T  ---->  GO TO 186
  F

  CALL DAYNXT(CDSDR,DPY)
186  CONTINUE

  IF VINTCR .LT. 0.25*VINTMR      T  ---->  VINTCR = VINTCR + DPET(
                                              DAY)/96.0
  F
```

```
187 ++++++++CCNTINUE
    +   C  END OF 15 MINUTE LOOP
    +
    +              IF
    +         CONOPT(5               T  ++++   GO TO 197
    +     *)  .NE. 1
    +           F
    +
    +   C  HOURLY OVERLAND FLOW AND RAINFALL SORTING
    +
    +              IF
    +         TOFR  .LE              T  ++++   GO TO 193
    +     *.  C.0
    +           F
    +
    +       | KT20 =20                                  |
    +
188 +
    +              IF
    +         KT20  .LT              T  ++++   GO TO 192
    +     *.  1
    +           F
    +
    +              IF
    +         TOFR  .GT              T  ++++   GO TO 189
    +     *.  T20OFH(KT20)
    +           F
    +
    +              |190  |
    +
189 +       T20OFH(KT20+1)  =  T20OFH(KT20)
    +              |191  |
    +
190 +       T20OFH(KT20+1)  =  TOFR
    +              |193  |
    +
191 +       | KT20 =KT20-1                              |
    +
    +              |188  |
    +
```

```
192   | T20PRH(1) =TOFR |

193   IF 0.0 .LE. PRH          T ---> | GO TO 197 |
      F

      | KT20 =20 |

194   IF 1 .LT. KT20           T ---> | GO TO 196 |
      F

      IF T20PRH(KT20) .GT. PRH T ---> | GO TO 195 |
      F

      T20PRH(KT20 + 1) = PRH

      |197 |

195   T20PRH(KT20+1) = T20PRH(KT20)

      | KT20 =KT20-1 |

      |194 |

196   | T20PRH(1) =PRH |

C  ADDING GROUNDWATER FLOW

197   | CBF  =GWS*BFRL*(1.0+BFNRL*BFNX) |
      | GWS  =GWS-CBF                   |
      | AMBF =AMBF+CBF                  |
      | THGR =ARHF+CBF                  |

      IF THGR .GT. HSE         T ---> | HSE = THGR |
      F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
AMSE =AMSE+HSE
THSF(HOUR) =(THGR-HSE)*WCFS
TDSF =TDSF+THSF(HOUR)

C

C STORE SIMULATED HOURLY STREAM FLOWS

C  DRAINING OF UPPER ZONE STORAGE

SINDEX =SINDEX+1

THSFD(SINDEX,3)=THSF(HOUR)

306  UZINFX =(UZS/UZC)-(LZS/LZC)

IF UZINFX .LE. 0.0    T -> GO TO 198
F

LZSR =LZS/LZC
UZINLZ =0.003*BMIR*UZC*UZINFX*
       *3.0

IF UZINLZ .GT. UZS    T -> UZINLZ = UZS
F

UZS  =UZS-UZINLZ
LZRX =1.5*ABS(LZSR-1.0)+1.0
FMR  =(1.0/(1.0+LZRX))**LZRX

IF LZS .LT. LZC    T -> FMR = 1.0 - FMR*LZSR
F

PGW  =(1.0-FMR)*UZINLZ*(1.0-SUBWF)
     *FPER
PLZS =FMR*UZINLZ
LZS  =LZS+PLZS
GWS  =GWS+PGW
BFNX =BFNX+PGW

C  4 PM ADJUSTMENTS OF VARIOUS VALUES
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
198 +
    +          . * IF * .        T   ---------------------------
    +    * HOUR .NE       * .  *++++| GO TO 202                 |
    +    *. 16                 *    |                           |
    +       * .       . *           ---------------------------
    +           F |
    +             |
    +    ----------------------------------------
    +    | AEX90=0.9*(AEX90+PET)                 |
    +    | AEX96=0.96*(AEX96+PET)                |
    +    ----------------------------------------
    +             |
    +  C  INFILTRATION CORRECTION
    +             |
    +    ----------------------------------------
    +    | SIAM =(AEX96/AETX)**SIAC              |
    +    ----------------------------------------
    +             |
    +          . * IF * .        T   ---------------------------
    +    * SIAM .LT       * .  *++++| SIAM = 0.33               |
    +    *. 0.33               *    |                           |
    +       * .       . *           ---------------------------
    +           F |                              +
    +             |++++++++++++++++++++++++++++++
    +    ----------------------------------------
    +    | BFNX =0.97*BFNX                       |
    +    ----------------------------------------
    +             |
    +          . * IF * .        T   ---------------------------
    +    * PET  .EQ.      * .  *++++| GO TO 202                 |
    +    * C.0                 *    |                           |
    +       * .       . *           ---------------------------
    +           F |
    +  C  EVAP-TRANS LOSS FROM GROUNDWATER
    +             |
    +    ----------------------------------------
    +    | GWET =GWS*GWETF*PET*FPER              |
    +    | GWS  =GWS-GWET                        |
    +    | BFNX =BFNX-GWET                       |
    +    ----------------------------------------
    +             |
    +          . * IF * .        T   ---------------------------
    +    * BFNX .LT       * .  *++++| BFNX = 0.0                |
    +    *. 0.0                *    |                           |
    +       * .       . *           ---------------------------
    +           F |                              +
    +             |++++++++++++++++++++++++++++++
    +    ----------------------------------------
    +    | AMPET=AMPET+PET                       |
    +    ----------------------------------------
    +             |
    +          . * IF * .        T   ---------------------------
    +    * PET  .GE.      * .  *++++| GO TO 199                 |
    +    * UZS                 *    |                           |
    +       * .       . *           ---------------------------
    +           F |
    +    ----------------------------------------
    +    | UZS  =UZS-PET                         |
    +    | AMNET=AMNET+PET                       |
    +    ----------------------------------------
    +             |
```

```
          202

199    PET   =PET-UZS
       AMNET=AMNET+UZS
       UZS   =0.0
       LZSR  =LZS/LZC

           IF
         PET .GE.                T
       ETLF*LZSR           *****     GO TO 200
           F

       SET   =PET*(1.0-PET/(2.0*ETLF*LZSR))

          201

200    SET   =0.5*ETLF*LZSR

201    LZS   =LZS-SET
       AMNET=AMNET+SET

202 ++++++++CONTINUE
     C  END OF HOUR LOOP

       DSSF(DAY)  =TDSF/24.0

           IF
        CONCPT(1                 T       DSSF(DAY) = DSSF(DAY) +
       1) .EQ. 1           *****        DDIW(DAY)
           F

203    AMRTF=AMRTF+DRSF(DAY)
       AMSTF=AMSTF+DSSF(DAY)

           IF
        CONOPT(6                 T
       ) .EQ. 1            *****         EDLZS(DAY) = LZS
           F

     C  STORE ERRORS AND FLOW DURATION
```

```
              IF                 T   GO TO 204
          CONOPT(4 ) .NE. 1
              F
    | ERR   =CSSF(DAY)-DRSF(DAY)                 |

              IF                 T   KRFMI = 1.0
          DRSF(DAY ) .LT. 1.0
              F

              IF                 T   KRFMI= 2.0*ALOG(DRSF(DA
          DRSF(DAY ) .GT. 1.0          Y)) + 2.0
              F

    | CRFMI(KRFMI)  =CRFMI(KRFMI)+1.0
    | SERR(KRFMI)  =SERR(KRFMI)+ERR
    | SERA(KRFMI)  =SERA(KRFMI)+ABS(ERR)
    | SQER(KRFMI)  =SQER(KRFMI)+ERR*ERR
    | SESF(KRFMI)  =0.0

              IF                 T   SESF(KRFMI) = SQRT(ABS(
          CRFMI(KR FMI) .GT. 1.0       (SQER(KRFMI) -
              F
      SERR(KRFMI)**2/CRFMI(KRFMI))/(CRFMI(KRFMI) - 1.0)))
204       CONTINUE
    | DATE =DAY                                   |

              IF                 T   DATE=MOD(DAY,MDAY)
          (MONTH.E Q.4).AND.(MDAY.EQ.3 1)
              F

              IF                 T   DATE=MOD(DAY,MDAY)
          MONTH.NE .4
              F

      WRITE(6,9)
      DATE, (THSF(HOUR),HOUR=1,12)
```

```
                                                      FORMAT      9 FORMAT(1H/,1X/,1X,I4,
                                                              2X,2HAM,1X,6F8.1,3X,6F8.1)
    +------------------------------------------------------+
   + WRITE(6,10)                                         +
  +  (THSF(HOUR),HOUR=13,24), DSSF(DAY)                 +
 +                                                     +
+-----------------------------------------------------+
                 |
                 |                                    FORMAT    10 FORMAT(1HJ,6X,2HPM,1X,
                 |                                             6F8.1,3X,7F8.1)
                 .                        ---------------------------------
          . * * * * .                    |                                 |
      . *     IF     * .  T              |    GO TO 205                    |
    *      TDFP24  .      *+++++         |                                 |
    *LT. 12.0             *              |                                 |
      * .          . *                    ---------------------------------
          * . * . *
            F |
              |
    ------------------------------
    | TDFP12 =TDFP24-12.0        |
    ------------------------------
              |
                                                      FORMAT     11 FORMAT(1H/,10X,8HMAXIMUM=
     4HP.M.)                                        ,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,
              |
           |206 |
           ------

205     CONTINUE
                                                      FORMAT     12 FORMAT(1H/,10X,8HMAXIMUM=
         4HA.M.)                                    ,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,
                                                      FORMAT  4033  FORMAT(1X,5HGFUS=,F7.3,1X,
              |                                               6HNRTRI=,F7.3,1X,5HRHFO=,F7.3)
206           .                         ---------------------------------
          . * * * * .                  |                                 |
      . *     IF     * .  T            |    WRITE(6,13)DATE,             |
    *    CONOPT(7        *+++++        |                                 |
    *) .EQ. 1 .AND. SDEP*              |                                 |
    *TH .GT. 0.0        *              |                                 |
      * .          . *                  ---------------------------------
          * . * . *
          F |+++++++++++++++++++++++++++++++++++++++++++++
    SDEPTH,STMD,SAX,TANSM,SPLW
                                                      FORMAT     13 FORMAT(3X,I4,2X,7HSDEPTH=
     2X,6HTANSM=,F6.2,2X,5HSPLW=,F6.2)              ,F8.2,2X,5HSTMD=,F6.2,2X,4HSAX=,F6.2,
              |
    ------------------------------
    | MAXI =MAXI+1               |
    | MPDAY(MAXI) =DATE          |
    ------------------------------
              |
  C  MONTHLY SUMMARY STORAGE

    STMROS(MAXI,3)=DSSF(DAY)
              |
    ------------------------------
    | TMSTF(MAXI,3) =AMSTF       |
    | AMSTF=0.0                  |
    | TMRTF(MONTH) =AMRTF        |
    | AMRTF=0.0                  |
    ------------------------------
              |
    EMBFNX(MAXI,3)=BFNX

    TMPREC(MAXI,3)=AMPREC
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                    |
 ---------------------------------------------
 | AMPREC =0.0                                |
 | TMRPM(MONTH) =AMRPM                        |
 | AMRPM=0.0                                  |
 | TMBF(MAXI,3) =AMBF                         |
 | AMBF =0.0                                  |
 | TMIF(MAXI,3) =AMIF                         |
 | AMIF =0.0                                  |
 | TMSE(MAXI,3) =AMSE                         |
 | AMSE =0.0                                  |
 | TMPET(MONTH) =AMPET                        |
 | AMPET=0.0                                  |
 | TMNET(MONTH) =AMNET                        |
 | AMNET=0.0                                  |
 | TMSNE(MONTH) =AMSNE                        |
 | AMSNE=0.0                                  |
 ---------------------------------------------
                    |
 TMFSIL(MONTH) = AMFSIL
                    |
 ---------------------------------------------
 | AMFSIL =0.0                                |
 | EMGWS(MAXI,3) =GWS                         |
 | UZC  =SUZC*AEX90+BUZC*EXP(-2.7             |
 |      *LZS/LZC)                             |
 ---------------------------------------------
                    |
            . * * .
       * IF     *   T      -------------------------
 *  UZC .LT.       *++++|   UZC = 0.25            |
 * 0.25            *    |                         |
       *  .    * *      -------------------------
           F |
             |+++++++++++++++++++++++++++++++
 ---------------------------------------------
 | EMUZC(MAXI,3) =UZC                         |
 | EMUZS(MAXI,3) =UZS                         |
 ---------------------------------------------
                    |
 EMSIAM(MAXI,3)=SIAM
                    |
 ---------------------------------------------
 | EMLZS(MAXI,3) =LZS                         |
 | EMIFS(MAXI,3) =IFS                         |
 ---------------------------------------------
                    |
220         CONTINUE
                    |
            . * * .
       *  IF    *   T      -------------------------
 *  MDAY.EQ.       *++++|   MDAY=59               |
 *337              *    |                         |
       *  .    * *      -------------------------
           F |
             |+++++++++++++++++++++++++++++++
 C
 C STORE MAXIMUM DAILY STREAM FLOW FOR YEAR
                    |
 ---------------------------------------------
 | TFMAXY(MAXI) =TFMAX                        |
 ---------------------------------------------
                    |
3001        CONTINUE
                    |
            . * * .
       *  IF    *   T      -------------------------
 *  DAY.EQ.E       *++++|   GO TO 3002            |
 *DATE             *    |                         |
       *  .    * *      -------------------------
           F |
```

```
          CALL DAYNXT(DAY,DPY)
                    |
                    .
                 .  *  *  .
          .  *     IF        *  .   T     +--------------------------------+
         *     DAY.GT.L             *++++ |        MDAY=LDAY               |
         *DAY                       *     |                                |
             *  .              .  *       +--------------------------------+
                 *  .  *
                 F  |                         +
                    |++++++++++++++++++++++++++++
                    |
                    .
                 .  *  *  .
          .  *     IF        *  .   T     +--------------------------------+
         *     DAY.EQ.3             *++++ |        MDAY=337                |
         *66                        *     |                                |
             *  .              .  *       +--------------------------------+
                 *  .  *
                 F  |                         +
                    |++++++++++++++++++++++++++++
                    |
               ---------
               |152    |
               ---------

 3002               |
                    .
                 .  *  *  .
          .  *     IF        *  .   T     +--------------------------------+
         *     CONOPT(1             *++++ |        GO TO 221               |
         *6).EQ.0                   *     |                                |
             *  .              .  *       +--------------------------------+
                 *  .  *
                 F  |

          BACKSPACE 11
                    |
          +---------------------------------------------------+
         +  WRITE (11)                                        +
        +   (RPLCTC(I),I=1,1832)                             +
       +                                                    +
      +---------------------------------------------------+
                    |

          BACKSPACE 18
                    |
          +---------------------------------------------------+
         +  WRITE (18)                                        +
        +   (RCOMMA(I),I=1,12087)                            +
       +                                                    +
      +---------------------------------------------------+
                    |

      C   END OF DAY LOOP

 221      CONTINUE
          RETURN

 END
```

```
SUBROUTINE PASTRN(PDAY,DATES,MONTH,EDATE,MSBDIC,PDATE,LDAY,MPDAY)

      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMRUS(121,6),DPY,TITLE,
      KFLAG,IDFLAG,
      IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
        NCTRI,CTRI,FIRR,RICY,DPSE,BDDFSM,SPBFLW,SPTWCC,SPM,FLDIF,
        XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,KGPMB,AREA,FIMP,
       SATRI,UHFA,
      MNRD,
        FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
        BIVF,OFSS,OFSL,OFMN,CFMNIS,IFRC,CSRX,FSRX,CHCAP,CXOPV,
        BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
        IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,FPAET,FPER,
        TPLR,VINTCR,HSE,NRTRI,SPIF,CBF,SPDR,OFUS,OFUSIS,OFR,OFRIS,PEIS,
        RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
        WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
        SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
      DEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
      TCMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
      TSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
      TCRARY(1,6,1),
      DRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)

      COMMCN/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
      TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
      EPCM,SERA,SERR,SESF,SQER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
      TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,
      TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
      TRHF,
      SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
C     KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)

C      BASED ON STANFORD WATERSHED MODELS III & IV

      DIMENSION BTRI(99),CONOPT(16),CRFMT(22),CTRI(99),DDIW(366),
        DMNT(366), DMXT(366), DPSE(366), DRGPM(366), DRHP(366,24),
        DRSGP(366), DPET(366), DRSF(366), DSSF(366), EDLZS(366),
       EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
       EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
        RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SQER(22),
       THSF(24),TITLE(18),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
       TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
       TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T20UFH(21),T20PRH(21),
       UHFA(99),TMRTFI(12),JULDAT(6),THSFD(744,3),TFMAXY(366),
       PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)

      LOGICAL  LSHFT

      INTEGER   CDSDR,CN,CONUPT,DATE,DAY,DPY,EHSGD,HOUR,HRF,HRL,PDAY,
        PRD,RHPD,RHPH,RSBD,SGMD,SGRT,SGRT2,YEAR,YR1,YR2,PHRS,SINDEX

      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY

      INTEGER  DATES,EDATE,SINDET,MPDAY(15)

      DIMENSION RPLOTC(1832),RCOMMA(12087)

      EQUIVALENCE (DPY,RPLOTC(1)),(CRFMI,RCOMMA(1))

      REAL  IFPRC,IFRC,IFRL,IFS,LZC,LZRX,LZS,LZSR,MHSM,MNRD,MRNSM,NHPT

      DATA  MEDCY/  0,        31,59,90,120,151,181,212,243,273,304,334/

      DATA  MEDWY/304,334,365,31,59,90,120,151,181,212,243,273        /

      REAL     MXDRSF,MXDSSF,MXMRSF,MXMSSF,SSQD

      REAL  SSQM,SSQDI,SSQMI,VDRSF,VDSSF

      REAL  VMRSF,VMSSF,SDDRSF,SDDSSF,SDMRSF,SDMSSF,SMDD,SMMD,SMSQD,
            SMSQM

      REAL  MDRSF,MDSSF,MMRSF,MMSSF

      BACKSPACE 11

       READ  (11)
       (RPLOTC(I),I=1,183…)
```

```
BACKSPACE 18

READ(18)
(RCOMMA(I),I=1,12087)

CALL READ(GWS,UZS,LZS,BFNX,IFS,UZC,SIAM)

WRITE (6,5005)
GWS,UZS,LZS,BFNX,IFS,UZC,SIAM

FORMAT 5005 FORMAT (1X,4HGWS=,F7.3,1X,4HUZS=,F7.3,1X,4HLZS=,F7.3,1X,5HBFNX=,
       F7.3,1X,4HIFS=,F7.3,1X,4HUZC=,F7.3,1X,5HSIAM=,F7.3)

150    CONTINUE

DATE =PDATE
DAY  =DATES
MDAY =PDAY
SINDEX =0
MAXI =0
AMRPM=0.0
AMPREC =0.0
AMBF =0.0
AMSE =0.0
AMSIF=0.0
AMRTF=0.0

   IF
DPY .EQ.  T   MEDWY(5)=366
366
 F

WRITE (6,4)
(TITLE(KTA),KTA=1,18)

                    FORMAT    4 FORMAT(1H1,10X,18A4,//)
                    FORMAT    3. FORMAT(1X,'PAST RUN HOURLY
                                   CFS VALUE',//,1X,4A4)

WRITE(6,3)
MSBCIC

C  BEGIN DAY LOOP

IMONTH =MONTH

152    TDSF =0.0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
              IF                      GO TO 148
          MONTH.NE              T
      .4                       *++++
              F

              IF                      GO TO 149
          MDAY.NE.              T
       31                      *++++
              F

148           IF                      IMONTH=MONTH+1
          DATE.GT.              T
      (MOD(DAY,MDAY))          *++++
              F  ++++++++++++++++++++++++++++++

149       CONTINUE

      PET  =EPCM(IMONTH)*UPET(DAY)
      PETU =PET
      TFMAX=0.0

C  EVAPOTRANSPIRATION ADJUSTMENTS

              IF                      GO TO 153
          CONOPT(7              T
      ) .NE. 1                 *++++
              F

              IF                      PET = 0.0
          DMXT(DAY              T
      ) - 4.0*ELDIF .LT.       *++++
      40.0
              F  ++++++++++++++++++++++++++++++

              IF                      PET = FFOR*PET
          SPTW .GT              T
      . SPTWCC                 *++++
              F  ++++++++++++++++++++++++++++++

C  CALCULATION OF SNOW EVAPORATION

              IF                      GO TO 153
          DMNT(DAY              T
      ) .GT. 32.0 .OR. SP      *++++
      TW .LE. DPSE(DAY)
              F
```

```
                 |
     ----------------------------------------------
     | SE    =DPSE(DAY)                           |
     | AMSNE=AMSNE+SE                             |
     | SPTW =SPTW-SE                              |
     ----------------------------------------------
                 |
                 .                  ---------------------------------
           . * * IF *  .    T      |     SDEPTH = SDEPTH - SE/SF    |
     * .  SFMD  .GT     * .  *+++++|   MD                           |
     *. 0.0                  *     |                                |
           *  .  *   .  *          ---------------------------------
               F |                          +
                 |+++++++++++++++++++++++++++
153         +-------------+
        +  DO LOOP TO     +
 ++++++++++ STMT #    202 +
 +      +  HOUR = 1,24    +
 +          +-------------+
 +               |
 +(NSGRD IF((NSGRD .EQ. 0) .AND. (DRHP(DAY,HOUR) .NE. 0.0)  .AND. (PET .EQ.
 +         PETU) .AND. (CCNOPT(3) .EQ. 1)) PET = 0.5*PET
 +
   (CONTINUED ON PAGE   5)
                                                               PAGE   5
 +
 +               |
154 +            .                  ---------------------------------
 +         . * * IF *  .    T      |     RGPM = DRGPM(DAY)          |
 +   *    HOUR  .EQ     * .  *+++++|                                |
 +   *. SGRT + 1             *     |                                |
 +         *  .  *   .  *          ---------------------------------
 +             F |                          +
 +               |+++++++++++++++++++++++++++
 +               .                  ---------------------------------
 +         . * * IF *  .    T      |     HSE = (FWTR*PET)/12.0      |
 +   *    HOUR  .EQ     * .  *+++++|                                |
 +   *. 9                    *     |                                |
 +         *  .  *   .  *          ---------------------------------
 +             F |                          +
 +               |+++++++++++++++++++++++++++
 +               .                  ---------------------------------
 +         . * * IF *  .    T      |     HSE = 0.0                  |
 +   *    HOUR  .EQ     * .  *+++++|                                |
 +   *.21                    *     |                                |
 +         *  .  *   .  *          ---------------------------------
 +             F |                          +
 +               |+++++++++++++++++++++++++++
 +     ----------------------------------------------
 +     | PRH   =RGPM*DRHP(DAY,HOUR)                 |
 +     | AMPREC =AMPREC+PRH                         |
 +     ----------------------------------------------
 +               |
 +  C  ENTER SNOWMELT SUBROUTINE
 +               |
 +               .                  ---------------------------------
 +         . * * IF *  .    T      |     CALL SNOMEL(BDDFSM,SPTW    |
 +   *    CONOPT(7      * .  *+++++|   CC,SPM,ELDIF,DAY,            |
 +   *) .EQ. 1               *     |                                |
 +         *  .  *   .  *          ---------------------------------
 +             F |                          +
 +               |+++++++++++++++++++++++++++
 +     SPBFLW, XDNFS,FFOR,FFSI,MRNSM,DSMGH,SDEPTH,STMD, ^XCSA,HOUR,
 +     SAX,SOFRF,OFRFIS,SOFRFI,AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,
 +     WT4AM,WT4PM,ASM,ASMRG,SASFX,SARAX,UMXT,DMNT,RICY,FIRR)
 +               |
 +     ----------------------------------------------
155 +  | AMRPM=AMRPM+PRH                            |
 +     ----------------------------------------------
```

```
156   | TOFR =0.0
      | ARHF =0.0

C  15 MINUTE ACCOUNTING AND ROUTING LOOP

      DO LOOP TO
      STMT #   187
      PRD = 1,4

      | PEBI =0.0 |

      | PPI   =0.0
      | OFR   =0.0
      | CFRIS=0.0
      | WI    =0.0
      | WEIFS=0.0
      | PMEUZS =0.0
      | PMELZS =0.0
      | PMEIFS =0.0
      | PMECFS =0.0
      | PEP   =0.25*PRH

      IF CONOPT(2) .EQ. 1    T   CALL PREPRD(RGPM,DRHP,DAY,HOUR,DPY,PRD,PEP,PRH)
      F

      IF PEP .GT. 0.0        T   GO TO 157
      F

      IF OFUS .GT. 0.0       T   GO TO 159
      F

      IF IFS .GT. 0.0        T   GO TO 170
      F

      IF NRTRI .GT. 0        T   GO TO 172
      F

      | TRHF =0.0 |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                IF                T     GO TO 181
          RHFO  .GT             *++++
    *. 0.0
                F
             |184 |

     C  RAINFALL UPPER ZONE INTERACTION

157             IF                T     GO TO 158
          PEP   .GE.            *++++
    * VINTCR
                F

     UZS    =UZS+PEP*TPLR
     VINTCR =VINTCR-PEP
     PPI    =0.0
     PEBI   =0.0
     PMEUZS =PEP

                IF                T     GO TO 159
          OFUS  .GT             *++++
    *. 0.0
                F
             |170 |

158  PPI    =PEP-VINTCR
     UZS    =UZS+VINTCR*TPLR
     VINTCR =0.0
     LZSR   =LZS/LZC
     UZC    =SUZC*AEX90+BUZC*EXP(-2.7
             *LZSR)

                IF                T     UZC = 0.25
          UZC   .LT.            *++++
    * 0.25
                F

     UZRX =2.0*ABS(UZS/UZC-1.0)+1.0
     FMR  =(1.0/(1.0+UZRX))**UZRX

                IF                T     FMR = 1.0 - FMR
          UZS   .GT.            *++++
    * UZC
                F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      | PEBI   =PPI*FMR
      | PMEUZS =PEP-PEBI
      | UZS    =UZS+PPI-PEBI

C   LOWER ZCNE AND GROUNDWATER INFILTRATION

159   | LZSR =LZS/LZC
      | EID  =4.0*LZSR

      IF LZSR .LE 1.0    T ---> GO TO 160
      F
      | EID  =4.0+2.0*(LZSR-1.0)

      IF LZSR .LE 2.0    T ---> GO TO 160
      F
      | EID  =6.0

160   | PEBI =PEBI+OFUS

      | CIMR =0.25*SIAM*BMIR/(2.0**EID)
      | CIVM =BIVF*2.0**LZSR

      IF CIVM .LT 1.0    T ---> CIVM = 1.0
      F

      | PEAI =PEBI*PEBI/(2.0*CMIR*CIVM)
      | WI   =PEBI*PEBI/(2.0*CMIR)

      IF PEBI .GE CMIR   T ---> WI = PEBI - 0.5*CMIR
      F

      IF PEBI .GE CMIR*CIVM   T ---> PEAI = PEBI - 0.5*CMIR*
      F                                 CIVM
```

```
        | WEIFS=WI-PEAI                                |

              IF                   T       GO TO 161
        PEBI .LE                 *++++
     CFUS
              F

        PMELZS =(PEBI-WI)*((PEBI-OFUS)
                /PEBI)
        PMEIFS =WEIFS*((PEBI-OFUS)/PEBI)
        PMEOFS =PEAI*((PEBI-OFUS)/PEBI)

161     CONTINUE

              IF                   T       GO TO 162
        (PEAI -                  *++++
     OFUS) .GT. 0.0
              F

        | EQD   =(OFUS+PEAI)/2.0                       |

              |163 |

162     | EQD   =EQDF*((PEAI-OFUS)**0.6)               |

163
              IF                   T       EQD = 0.5*(OFUS + PEAI)
        (OFUS +                  *++++
     PEAI) .GT. (2.0*EQD*
     *)
              F  +++++++++++++++++++++++++++++++

              IF                   T       GO TO 164
        (OFUS +                  *++++
     PEAI) .LE. 0.001
              F

        OFR   =0.25*OFRF*(((OFUS+PEAI)*0.5)
               **1.67)*((1.0+0.6*((OFUS
               +
               PEAI)/(2.0*EQD))**3.0)**1.67)

              IF                   T       OFR = 0.75*PEAI
        OFR .GT.                 *++++
     (0.75*PEAI)
              F  +++++++++++++++++++++++++++++++
```

```
164 +               |
    +          . * . * .            ---------------------------------
    +      . *    IF     * .   T   |     GO TO 168                   |
    +    *   FIMP .EQ       *+++++ |                                 |
    +    *. 0.0             *      |                                 |
    +      * .         . *          ---------------------------------
    +          * . *
    +          F  |
    +             |
    +  -----------------------------------------------
165 +  | PEIS =PPI+OFUSIS                             |
    +  -----------------------------------------------
    +             |
    +          . * . * .            ---------------------------------
    +      . *    IF     * .   T   |     GO TO 166                   |
    +    *     (PEIS -      *+++++ |                                 |
    +    *OFUSIS) .GT. 0.0  *      |                                 |
    +      * .         . *          ---------------------------------
    +          * . *
    +          F  |
    +             |
    +  -----------------------------------------------
    +  | EQDIS=(OFUSIS+PEIS)/2.0                      |
    +  -----------------------------------------------
    +             |
    +          ---------
    +          |167   |
    +          ---------
    +
    +             |
    +  -----------------------------------------------
166 +  | EQDIS=EQDFIS*((PEIS-OFUSIS)**0.6)            |
    +  -----------------------------------------------
    +             |
167 +             |
    +          . * . * .            ---------------------------------
    +      . *    IF     * .   T   |     EQDIS = 0.5*(OFUSIS + P     |
    +    *     (OFUSIS      *+++++ |   EIS)                          |
    +    *+ PEIS) .GT. (2.0*E*     |                                 |
    +    *QDIS)             *      |                                 |
    +      * .         . *          ---------------------------------
    +          * . *                         +
    +          F  |++++++++++++++++++++++++++++++
    +             |
    +          . * . * .            ---------------------------------
    +      . *    IF     * .   T   |     GO TO 168                   |
    +    *     (OFUSIS      *+++++ |                                 |
    +    *+ PEIS) .LE. 0.01 *      |                                 |
    +      * .         . *          ---------------------------------
    +          * . *
    +          F  |
    +             |
    +  -----------------------------------------------
    +  | OFRIS=0.25*OFRFIS*(((OFUSIS+PEIS)            |
    +  |         *0.5)**1.67)*((1.0+0.6*((            |
    +  |           OFUSIS + PEIS)/(2.0*EQDFIS))**3.0)**1.67)
    +  -----------------------------------------------
    +             |
    +          . * . * .            ---------------------------------
    +      . *    IF     * .   T   |     OFRIS = PEIS                |
    +    *   OFRIS .G       *+++++ |                                 |
    +    *T. PEIS           *      |                                 |
    +      * .         . *          ---------------------------------
    +          * . *                         +
    +          F  |++++++++++++++++++++++++++++++
    +             |
    +  -----------------------------------------------
168 +  | TOFR =TOFR+FPER*OFR+FIMP*OFRIS               |
    +  |        +PPI*FWTR                             |
    +  | OFUSIS =PEIS-OFRIS                           |
    +  | OFUS =PEAI-OFR                               |
    +  -----------------------------------------------
    +             |
```

```
         +                   !
         +                   .                           ----------------------------------
         +               . * . * .                       |                                |
         +          . *      IF       * .   T            |    GO TO 169                   |
         +        *     OFUS  .GE          *****+        |                                |
         +        *. C.C01                 *             |                                |
         +           * .            . *                  |                                |
         +               * . *                           ----------------------------------
         +                F  |
         +                   |
         +                   |
         +        ----------------------------------------
         +        | LZS   =LZS+OFUS                      |
         +        | OFUS  =0.0                           |
         +        | OFRIS=OFRIS+OFUSIS                   |
         +        | OFUSIS =0.0                          |
         +        ----------------------------------------
         +                   |
         +        ----------------------------------------
169      +        | LZRX  =1.5*ABS(LZS/LZC-1.0)+1.0      |
         +        | FMR   =(1.0/(1.0+LZRX))**LZRX        |
         +        ----------------------------------------
         +                   |
         +                   |
         +                   .                           ----------------------------------
         +               . * . * .                       |                                |
         +          . *      IF       * .   T            |   FMR  =  1.0  - FMR*(LZS/LZ   |
         +        *     LZS  .LT.          *****+        | C)                             |
         +        * LZC                    *             |                                |
         +           * .            . *                  |                                |
         +               * . *                           ----------------------------------
         +                F  |                                        +
         +                   |+++++++++++++++++++++++++++++++++++++++++
         +        ----------------------------------------
         +        | PLZS  =FMR*(PEBI-WI)                 |
         +        | PGW   =(1.0-FMR)*(PEBI-WI)*(1.0      |
         +        |        -SUBWF)*FPER                  |
         +        | GWS   =GWS+PGW                       |
         +        | BFNX  =BFNX+PGW                      |
         +        | LZS   =LZS+PLZS                      |
         +        | IFS   =IFS+WEIFS*FPER                |
         +        ----------------------------------------
         +                   |
         +        ----------------------------------------
170      +        | SPIF  =IFRL*IFS                      |
         +        | AMIF  =AMIF+SPIF                     |
         +        | IFS   =IFS-SPIF                      |
         +        ----------------------------------------
         +                   |
         +                   |
         +                   .                           ----------------------------------
         +               . * . * .                       |                                |
         +          . *      IF       * .   T            |    GO TO 171                   |
         +        *     IFS  .GE.          *****+        |                                |
         +        * C.0001                 *             |                                |
         +           * .            . *                  |                                |
         +               * . *                           ----------------------------------
         +                F  |
         +                   |
         +        ----------------------------------------
         +        | LZS   =LZS+IFS                       |
         +        | IFS   =0.0                           |
         +        ----------------------------------------
         +                   |
         +        ----------------------------------------
171      +        | UHFA(1)  =FPER*OFR+PPI*FWTR+FIMP     |
         +        |           *OFRIS+SPIF                |
         +        | SPDR  =UHFA(1)                       |
         +        ----------------------------------------
         +                   |
         +   C   ROUTING
         +                   !
172      +                   .                           ----------------------------------
         +               . * . * .                       |                                |
         +          . *      IF       * .   T            |    GO TO 173                   |
         +        *     CONOPT(1           *****+        |                                |
         +        *2)  .NE. 1              *             |                                |
         +           * .            . *                  |                                |
         +               * . *                           ----------------------------------
         +                F  |
         +                   |
         +        ----------------------------------------
         +        | URHF  =URHF+0.25*UHFA(1)             |
         +        ----------------------------------------
         +                   |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
         IF
      PRD .NE.          T
     4               *++++  GO TO 181
         F

   UHFA(1) =URHF

173  TRHF =0.0
     KTRI =NCTRI

         IF
      CONOPT(1          T
     3) .EQ. 1       *++++  KTRI = NCSTRI
         F

174  URHF =UHFA(KTRI)

         IF
      URHF .LE          T
     0.0             *++++  GO TO 176
         F

175  TRHF =TRHF+URHF*CTRI(KTRI)

         IF
      CONOPT(1          T
     3) .EQ. 1 .AND. LSH*++++  TRHF = TRHF +
     FT .AND. KTRI .GE.
         F
     URHF*SATRI(KTRI - 1)
     UHFA(KTRI + 1) = URHF

          177

176  UHFA(KTRI+ 1) = 0.0
177  KTRI =KTRI-1

         IF
      KTRI .GE          T
     1               *++++  GO TO 174
         F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
178   IF URHF .LE. C.0                         T → GO TO 179
      F
      NRTRI=NCTRI

      IF CONOPT(13) .EQ. 1                     T → NRTRI = MXTRI
      F

179   NRTRI=NRTRI-1
      UHFA(1) =0.0

      IF CONOPT(13) .NE. 1                     T → GO TO 180
      F
      NNSTRI =NCSTRI+1
      UHFA(NNSTRI) =0.0

180   URHF =0.0

181   IF SRX .LE. CSRX                         T → SRX = CSRX
      F

      RHF1 =TRHF-SRX*(TRHF-RHFO)
      RHFO =RHF1

      IF RHFO .LT. RHFMC                       T → RHFO = 0.0
      F

      TFCFS=(4.0*RHF1+CBF-HSE)*WCFS

      IF CONOPT(13) .NE. 1                     T → GO TO 182
      F
```

```
         |
    . * * * .
  . *   IF    * .  T  ------------------------------
 *   CONOPT(1    *+++++|    GO TO 182               |
 *2) .EQ. 1 .AND. PRD*  |                            |
 * .NE. 4            *  |                            |
    * .         . *     ------------------------------
        *   *
        F |
CALL RTVARY (CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCSTRI,EXQPV,LSHFT,
  TFCFS)
          |
-----------------------------------------------
| CATE =MOD(DAY,MDAY)                          |
-----------------------------------------------
          |
    . * * * .            ------------------------------
  . *   IF    * .  T     |    WRITE(6,6) DATE,HOUR,PR  |
 *  LSHFT        *+++++  |  D,NCSTRI                  |
    * .         . *      ------------------------------
        *   *
        F |++++++++++++++++++++++++++++++++
                                 FORMAT     6 FORMAT(2X,I2,2X,I2,2X,I2
  8HELEMENTS)                   ,2X,20HHISTOGRAM CHANGES TO,1X,I2,1X,
182  CONTINUE
          |
    . * * * .            ------------------------------
  . *   IF    * .  T     |    SRX = CSRX               |
 *  TFCFS .L     *+++++  |                            |
 *E. 0.5*CHCAP   *       |                            |
    * .         . *      ------------------------------
        F |++++++++++++++++++++++++++++++++
          |
    . * * * .            ------------------------------
  . *   IF    * .  T     |    SRX = CSRX               |
 *  (TFCFS .     *+++++  |                            |
 *GT. 0.5*CHCAP) .AND*   |                            |
 *. (TFCFS .LT. 2.0*C*   |                            |
    * .         . *      ------------------------------
        F |++++++++++++++++++++++++++++++++
  +(FSRX - CSRX)*((TFCFS - 0.5*CHCAP)/(1.5*CHCAP))**3
          |
    . * * * .            ------------------------------
  . *   IF    * .  T     |    SRX = FSRX               |
 *  TFCFS .G     *+++++  |                            |
 *T. 2.0*CHCAP   *       |                            |
    * .         . *      ------------------------------
        F |++++++++++++++++++++++++++++++++
          |
    . * * * .            ------------------------------
  . *   IF    * .  T     |    GO TO 183                |
 *  TFCFS .L     *+++++  |                            |
 *E. TFMAX       *       |                            |
    * .         . *      ------------------------------
        F |
          |
-----------------------------------------------
| PRDF =PRD                                    |
| TDFP24 =HOUR                                 |
-----------------------------------------------
          |
    . * * * .            ------------------------------
  . *   IF    * .  T     |    TDFP24 = (TDFP24 - 1.C)  |
 *  PRD .LE.     *+++++  |  + 0.15*PRDF               |
 * 3             *       |                            |
    * .         . *      ------------------------------
        F |++++++++++++++++++++++++++++++++
```

```
        --------------------------------------
        | TFMAX=TFCFS                        |
        --------------------------------------
183     | ARHF =ARHF+RHF1                    |
        --------------------------------------

    C   STORM OUTPUT REQUESTED BY CONOPT(1)

184
              IF                    T
          CONOPT(1                 *+++++   GO TO 186
      *) .NE. 1
              F

              IF                    T
          DAY .NE.                 *+++++   GO TO 186
      * CDSDR
              F

              IF                    T
          HOUR .EQ                 *+++++   WRITE(6,7)
      *. 1 .AND. PRD .EQ.
      *1
              F |+++++++++++++++++++++++++++++
                                   FORMAT     7 FORMAT(1H//,21X,19HRAINFALL
                                                DEPOSITION,12X,16HMOISTURE STORAGE,
      14X,17HSTREAMFLOW ORIGIN,6X,14HSTREAM OUTFLOW/2X,116HDY HR PD RA
    IN    EUZS    ELZS    EIFS    EOFS      UZS     LZS     IFS     OFS     S
    PCF    SPIF    SPBF    SPTF   INCHES     CFS)

        --------------------------------------
        | DATE =MOD(DAY,MDAY)                |
        --------------------------------------

        --------------------------------------
        | OFS  =OFUS*FPER+OFUSIS*FIMP        |
        | SPCF =OFR*FPER+OFRIS*FIMP+PPI*FWTR |
        | SPBF =0.25*(CBF-HSE)               |
        | SPTF =SPDR+SPBF                    |
        | SPDR =0.0                          |
        --------------------------------------

              IF                    T
          RHFO .LE                 *+++++   TFCFS = (CBF - HSE)*WCF
      *. 0.0                                S
              F |+++++++++++++++++++++++++++++

        --------------------------------------
        | RSPTF=0.25*TFCFS/WCFS              |
        --------------------------------------

       WRITE(6,8)
       DATE,HOUR,PRD,PEP,PMEUZS,PMELZS,PMEIFS,PMEOFS,U+
     +ZS,LZS

     ,IFS,OFS,SPOF,SPIF,SPBF,SPTF,RSPTF,TFCFS
                                   FORMAT     8 FORMAT(2X,I2,1X,I2,1X,I1
      F7.1)                        ,5(1X,F6.4),2X,4(F7.4),2X,5(1X,F6.4),1X,
```

```
          IF HOUR .EQ. 24 .AND. PRD .EQ. 4          T    GO TO 185
          F
          |186 |

185       NDSCP=NDSDP+1

          IF NDSDR .EQ. NDSDP                       T    GO TO 186
          F
          CALL DAYNXT(CDSDR,DPY)
186       CCNTINUE

          IF VINTCR .LT. 0.25*VINTMR                T    VINTCR = VINTCR + DPET(
                                                         DAY)/96.0
          F
187       CCNTINUE
     C  END OF 15 MINUTE LOOP

          IF CONOPT(5) .NE. 1                       T    GO TO 197
          F
     C  HOURLY OVERLAND FLOW AND RAINFALL SORTING

          IF TOFR .LE. 0.0                          T    GO TO 193
          F
          KT20 =20
188
          IF KT20 .LT. 1                            T    GO TO 192
          F
```

```
                 |
             . * . * .
        . *     IF      * .   T    ------------------------------
      *      TOFR  .GT         *++++ |   GO TO 189                |
      *.  T200FH(KT20)         *     |                            |
        *  .     *   .  *            ------------------------------
              *  .  *
              F  |
                 |
             ---------
             |190    |
             ---------

189          T200FH(KT20+1)  =  T200FH(KT20)
                 |
             ---------
             |191    |
             ---------

190          T200FH(KT20+1)  =  TOFR
                 |
             ---------
             |193    |
             ---------

                 |
             ------------------------------------------
191          | KT20 =KT20-1                           |
             ------------------------------------------
                 |
             ---------
             |188    |
             ---------

                 |
             ------------------------------------------
192          | T200FH(1) =TOFR                        |
             ------------------------------------------
                 |
193          . * . * .
        . *     IF      * .   T    ------------------------------
      *      PRH   .LE.        *++++ |   GO TO 197                |
      *  0.0                   *     |                            |
        *  .     *   .  *            ------------------------------
              *  .  *
              F  |
                 |
             ------------------------------------------
             | KT20 =20                               |
             ------------------------------------------
                 |
194          . * . * .
        . *     IF      * .   T    ------------------------------
      *      KT20  .LT         *++++ |   GO TO 196                |
      *.  1                    *     |                            |
        *  .     *   .  *            ------------------------------
              *  .  *
              F  |
                 |
             . * . * .
        . *     IF      * .   T    ------------------------------
      *      PRH   .GT.        *++++ |   GO TO 195                |
      *  T20PRH(KT20)          *     |                            |
        *  .     *   .  *            ------------------------------
              *  .  *
              F  |
                 |
             T20PRH(KT20 + 1)  =  PRH
                 |
```

ORIGINAL PAGE IS OF POOR QUALITY

```
          |
       -------
       |197  |
       -------

195       T20PRH(KT20+1) = T20PRH(KT20)
                  |
          ---------------------------------------------
          | KT20 =KT20-1                              |
          ---------------------------------------------
                  |
       -------
       |194  |
       -------

                  |
          ---------------------------------------------
196       | T20PRH(1) =PRH                            |
          ---------------------------------------------
                  |
     C  ADDING GROUNDWATER FLOW
                  |
          ---------------------------------------------
197       | CBF  =GWS*BFRL*(1.0+BFNRL*BFNX)           |
          | GWS  =GWS-CBF                             |
          | AMBF =AMBF+CBF                            |
          | THGR =ARHF+CBF                            |
          ---------------------------------------------
                  |
                                        ------------------------------
           *  *  IF  *  *   T           |                            |
        *  HSE  .GT.       *++++        |     HSE = THGR             |
        *  THGR           *             |                            |
           *  *     *  *                ------------------------------
               F  |                                   +
                  |++++++++++++++++++++++++++++++++++++
          ---------------------------------------------
          | AMSE =AMSE+HSF                            |
          | THSF(HOUR) =(THGR-HSE)*WCFS               |
          | TDSF =TDSF+THSF(HOUR)                     |
          ---------------------------------------------
                  |
     C

     C STORE SIMULATED HOURLY STREAM FLOWS....

     C  DRAINING OF UPPER ZONE STORAGE
                  |
          ---------------------------------------------
          | SINDEX =SINDEX+1                          |
          ---------------------------------------------
                  |
          THSFD(SINDEX,1)=THSF(HOUR)
                  |
          ---------------------------------------------
306       | UZINFX =(UZS/UZC)-(LZS/LZC)               |
          ---------------------------------------------
                  |
                                        ------------------------------
           *  *  IF  *  *   T           |                            |
        *    UZINFX  .     *++++        |     GO TO 198              |
        *LE. 0.0          *             |                            |
           *  *     *  *                ------------------------------
               F  |
                  |
          ---------------------------------------------
          | LZSR =LZS/LZC                             |
          | UZINLZ =0.003*BMIR*UZC*UZINFX*            |
          |      *3.0                                 |
          ---------------------------------------------
```

```
      IF UZINLZ .GT. UZS                    T    UZINLZ = UZS
      F

      UZS  =UZS-UZINLZ
      LZRX =1.5*ABS(LZSR-1.0)+1.0
      FMR  =(1.0/(1.0+LZRX))**LZRX

      IF LZS .LT. LZC                       T    FMR = 1.0 - FMR*LZSR
      F

      PGW  =(1.0-FMR)*UZINLZ*(1.0-SUBWF)
            *FPER
      PLZS =FMR*UZINLZ
      LZS  =LZS+PLZS
      GWS  =GWS+PGW
      BFNX =BFNX+PGW

C   4 PM ADJUSTMENTS OF VARIOUS VALUES

198   IF HOUR .NE. 16                       T    GO TO 202
      F

      AEX90=0.9*(AEX90+PET)
      AEX96=0.96*(AEX96+PET)

C   INFILTRATION CORRECTION

      SIAM =(AEX96/AETX)**SIAC

      IF SIAM .LT. 0.33                     T    SIAM = 0.33
      F

      BFNX =0.97*BFNX

      IF PET .EQ. 0.0                       T    GO TO 202
      F
```

```
    C  EVAP-TRANS LOSS FRCM GROUNDWATER

       | GWET  =GWS*GWETF*PET*FPER   |
       | GWS   =GWS-GWET             |
       | BFNX  =BFNX-GWET            |

            IF
       BFNX .LT                 T
       0.0                 *****   |   BFNX = 0.0   |
          F

       | AMPET=AMPET+PET             |

            IF
       PET  .GE.                T
       UZS                 *****   |   GO TO 199    |
          F

       | UZS   =UZS-PET              |
       | AMNET=AMNET+PET             |

       |202 |

199    | PET   =PET-UZS              |
       | AMNET=AMNET+UZS             |
       | UZS   =0.0                  |
       | LZSR  =LZS/LZC              |

            IF
       PET  .GE.                T
       ETLF*LZSR           *****   |   GO TO 200    |
          F

       | SET   =PET*(1.0-PET/(2.0*ETLF*LZSR))   |

       |201 |

200    | SET   =0.5*ETLF*LZSR        |

201    | LZS   =LZS-SET              |
       | AMNET=AMNET+SET             |

202 ********CCNTINUE
```

```
C   END OF HOUR LOOP .

      | DSSF(DAY) =TDSF/24.0 |

      IF CONOPT(11) .EQ. 1   T ---> | DSSF(DAY) = DSSF(DAY) + DDIW(DAY) |
      F

203   | AMRTF=AMRTF+DRSF(DAY)
      | AMSTF=AMSTF+DSSF(DAY) |

      IF CONOPT(6) .EQ. 1   T ---> | EDLZS(DAY) = LZS |
      F

C   STORE ERRORS AND FLOW DURATION

      IF CONOPT(4) .NE. 1   T ---> | GO TO 204 |
      F

      | ERR  =DSSF(DAY)-DRSF(DAY) |

      IF DRSF(DAY) .LT. 1.0   T ---> | KRFMI = 1.0 |
      F

      IF DRSF(DAY) .GT. 1.0   T ---> | KRFMI= 2.0*ALOG(DRSF(DAY)) + 2.0 |
      F

      | CRFMI(KRFMI) =CRFMI(KRFMI)+1.0
      | SERR(KRFMI)  =SERR(KRFMI)+ERR
      | SERA(KRFMI)  =SERA(KRFMI)+ABS(ERR)
      | SQER(KRFMI)  =SQER(KRFMI)+ERR*ERR
      | SESF(KRFMI)  =0.0 |

      IF CRFMI(KRFMI) .GT. 1.0   T ---> | SESF(KRFMI) = SQRT(ABS.
                                          (SQER(KRFMI) -
      F
        SERR(KRFMI)**2/CRFMI(KRFMI))/(CRFMI(KRFMI) - 1.0)))
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
204       CONTINUE

          | DATE =DAY                                    |

          IF (MCNTH.EQ.4).AND.(MDAY.EQ.31)    T ++++   DATE=MOD(DAY,MDAY)
          F

          IF MONTH.NE.4                       T ++++   DATE=MOD(DAY,MDAY)
          F

          WRITE(6,9)
          DATE, (THSF(HOUR),HOUR=1,12)

                                    FORMAT     9 FORMAT(1H/,1X/,1X,I4,2X,
                                               2HAM,1X,6F8.1,3X,6F8.1)

          WRITE(6,10)
          (THSF(HOUR),HCUR=13,24), DSSF(DAY)

                                    FORMAT    10 FORMAT(1HJ,6X,2HPM,1X,
                                               6F8.1,3X,7F8.1)

          IF TDFP24 .LT. 12.0                 T ++++   GO TO 205
          F

          | TDFP12 =TDFP24-12.0                          |

                                    FORMAT    11 FORMAT(1H/,10X,8HMAXIMUM=
                                            ,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,
            4HP.M.)

               |206 |

205       CONTINUE
                                    FORMAT    12 FORMAT(1H/,10X,8HMAXIMUM=
                                            ,F8.1,2X,6HC.F.S.,5X,4HTIME,3X,F5.2,2X,4HA.M.)

206
          IF CONOPT(7) .EQ. 1 .AND. SDEPTH .GT. 0.0   T ++++   WRITE(6,13)DATE,
          F
          SDEPTH,STMD,SAX,TANSM,SPLW
                                    FORMAT    13 FORMAT(3X,I4,2X,7HSDEPTH=
                                            ,F8.2,2X,5HSTMD=,F6.2,2X,4HSAX=,F6.2,
                                            2X,6HTANSM=,F6.2,2X,5HSPLW=,F6.2)
```

```
              |
     ------------------------------------------------
     | MAXI =MAXI+1                                 |
     | MPDAY(MAXI) =DATE                            |
     ------------------------------------------------
              |
C  MONTHLY SUMMARY STORAGE

     STMROS(MAXI,1)=DSSF(DAY)
              |
     ------------------------------------------------
     | TMSTF(MAXI,1) =AMSTF                         |
     | AMSTF=C.0                                    |
     | TMRTF(MONTH) =AMRTF                          |
     | AMRTF=0.0                                    |
     ------------------------------------------------
              |
     EMBFNX(MAXI,1)=BFNX

     TMPREC(MAXI,1)=AMPREC
              |
     ------------------------------------------------
     | AMPREC =0.0                                  |
     | TMRPM(MONTH) =AMRPM                          |
     | AMRPM=0.0                                    |
     | TMBF(MAXI,1) =AMBF                           |
     | AMBF =0.0                                    |
     | TMIF(MAXI,1) =AMIF                           |
     | AMIF =0.0                                    |
     | TMSE(MAXI,1) =AMSE                           |
     | AMSE =0.0                                    |
     | TMPET(MONTH) =AMPET                          |
     | AMPET=0.0                                    |
     | TMNET(MONTH) =AMNET                          |
     | AMNET=0.0                                    |
     | TMSNE(MONTH) =AMSNE                          |
     | AMSNE=0.0                                    |
     ------------------------------------------------
              |
     TMFSIL(MONTH) = AMFSIL
              |
     ------------------------------------------------
     | AMFSIL =0.0                                  |
     | EMGWS(MAXI,1) =GWS                           |
     | UZC   =SUZC*AEX90+BUZC*EXP(-2.7              |
     |       *LZS/LZC)                              |
     ------------------------------------------------
              |
           * . *          --------------------------
     . *    IF    * .  T  |                        |
   *    UZC .LT.       *++++|    UZC = 0.25          |
   * 0.25              *  |                        |
     * .         . *      --------------------------
           * . *
          F |++++++++++++++++++++++++++++++++++++++++
              |
     ------------------------------------------------
     | EMUZC(MAXI,1) =UZC                           |
     | EMUZS(MAXI,1) =UZS                           |
     ------------------------------------------------
              |
     EMSIAM(MAXI,1)=SIAM
              |
     ------------------------------------------------
     | EMLZS(MAXI,1) =LZS                           |
     | EMIFS(MAXI,1) =IFS                           |
     ------------------------------------------------
              |
220  CONTINUE
              |
           * . *          --------------------------
     . *    IF    * .  T  |                        |
   *   MDAY.EQ.        *++++|    MDAY=59             |
   *337                *  |                        |
     * .         . *      --------------------------
           * . *
          F |++++++++++++++++++++++++++++++++++++++++

C
```

```
C STORE MAXIMUM DAILY STREAM FLOW FOR YEAR

        TFMAXY(MAXI) =TFMAX

3001    CONTINUE

        IF DATE DAY.EQ.E                T    GO TO 3002
        F

        CALL DAYNXT(DAY,DPY)

        IF DAY DAY.GT.L                 T    MDAY=LDAY
        F

        IF 66 DAY.EQ.3                  T    MDAY=337
        F

        152

3002    IF CONOPT(16).EQ.0              T    GO TO 221
        F

        BACKSPACE 11

        WRITE (11)
        (RPLCTC(I),I=1,1832)

        BACKSPACE 18

        WRITE (18)
        (RCCMMA(I),I=1,12087)

C  END OF DAY LOOP

221     CONTINUE
        RETURN

END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE OUTPUT(DIFFS,DIFFMP,DIFFP,DIFFPP,SUM,DIFFR,DIFFRP,
                        APREC,DIFFPR,DIFPP,QOUT,QMAX,PHRO,MSBDIC,
                        MPDAY)
      COMMON/PLUTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
      KFLAG,IDFLAG,
      IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
        NCTRI,CTRI,FIRR,RICY,DPSE,BDUFSM,SPBFLW,SPTWCC,SPM,ELDIF,
        XUNFS,FFCR,FFSI,MRNSM,DSMGH,PXCSA,KMPF,RGPMB,AREA,FIMP,
       SATRI,UHFA,
      MNRD,
        FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
        BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXQPV,
        BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNKL,BFNHR,IFPRC,
        IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BTRI,TFCFS,EPAET,FPER,
        TPLR,VINTCR,HSE,NRTRI,SPIF,CSF,SPDR,OFUS,OFUSIS,OFF,OFRIS,PEIS,
        RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
        WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQOF,EQCFIS,SOFRF,SOFRFI,
        SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
      DEPEND(2),VARIN(2),NPTS,JULDI,IYR,TODARY(5,1),
      TOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
      TSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
      TORARY(1,6,1),
      DRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)

      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
      TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
      EPCM,SERA,SERR,SESF,SWER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
      TMSTFI,T20OFH,T20PRH,TMRTFI,JULDAT,
      TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
      TRHF,
      SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS
C  KENTUCKY WATERSHED MODEL (VERSION OF JUNE 6, 1970)
C    BASED ON STANFORD WATERSHED MODELS III & IV
      DIMENSION BTRI(99),CONUPT(16),CRFMI(22),CTRI(99),DDIW(366),
        DMNT(366), DMXT(366), DPSF(366), DRGPM(366), DRHP(366,24),
        DRSGP(366), DPET(366), DPSF(366), DSSF(366), EDLZS(366),
       EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
       EMUZC(15,3),EMUZS(15,3),EPCM(12),FIRR(15),MEDCY(12),MEDWY(12),
        RICY(37), SATRI(99), SERA(22), SERR(22), SESF(22), SWER(22),
       THSF(24),TITLE(13),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
       TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMRTF(12),TMSE(15,3),
       TMSNE(12),TMSTF(15,3),TMSTFI(15,3),T20OFH(21),T20PRH(21),
       UHFA(99),TMRTFI(12),JULDAI(6),THSFD(744,3),TFMAXY(366),
       PEAKS(6),PHRS(6),NSPTS(6),THSFDM(6)
      REAL*8 MSVT(9)/'MOISTURE',' STORAGE',' VALUES ',
     '       T','ABLE 3A ',4*'        '/
      REAL*8 SUBT(11)/'ABLE 3A','ABLE 3B','    PAST',' 3A     ','UES
     ','PAST RUN','ABLE 1A ','ABLE 1B ',' 3B     ','MMARY   ',
     '    PAST '/
      REAL QOUT(15),SUM(3),APREC(3),MSBDIC
      INTEGER MPDAY(15),TYPE,CONOPT
      REAL*4 SCPAST(36)/18*'    ',3*'    ','OBSE','RVED','    ',
     'SIMU','LATE','D   ',3*'    ','OBSE','RVED','    ','SIMU','LATE',
     'D   '/
      REAL*4 MSPAST(36)/20*'    ','SIMU','LATE','D   ',5*'    ','SIMU',
     'LATE','D   ',5*'    '/
      REAL*4 SSPAST(36)/21*'    ','   O','BSER','VED ','   S','IMUL',
     'ATED','    ','DIFF','    ','%DIF','F   ',4*'    '/
      REAL*8 STST(9)/'STORM SU','MMARY TA','BLE 2   ',6*'        '/
      REAL*8    STRMT(9)/'STORM CH','ARACTERI','STICS  T','ABLE 1A ',
     5*'        '/
      REAL*8    STCHAR(252)/'        ','STORM CH','ARACTERI','STICS   '
     ,'FORECAST',' RUN   T','ABLE 1A ',12*'        ','     WORS',
     'T   NO  ','    F/CA','ST      ','        ','WORST   ','NO      ',
     'F/CAST  ','        ','    CASE','    PREC','IP      ',
     2*'        ','CASE    ','PRECIP  ',9*'        ','        ',
     '    PREC',
     'IP      ',3*'        ','PRECIP  ',3*'        ','        ','  SUR ',
     197*'        '/
```

```
REAL*8 STCSP(197)/'R/O        ',
3*'          ','SUR R/O ',3*'          ','     INT ',
'FL        ',3*'          ','INT FL  ',3*'          ','    BASE',
' FL       ',3*'          ','BASE FL ',3*'          ','    STM ',
'R/O       ',3*'          ','STM R/O ',3*'          ',9*'     ','
PREC','IP        ',3*'         ','PRECIP  ',3*'        ','     SUR ',
'R/O       ',3*'         ','SUR R/O ',3*'          ','     INT ','FL
  ',3*'         ','INT FL  ',3*'        ','     BASE',' FL      ',3*'
        ','BASE FL ',3*'          ','     STM ','R/O      ',3*'
  ','STM R/O ',3*'         ',9*'        ','     PREC','IP       ',3*'
        ','PRECIP  ',3*'          ','     SUR ','R/O      ',3*'
','SUR R/O ',3*'         ',
'       INT ','FL      ',3*'          ','INT FL  ',3*'        ','
BASE',' FL        ',3*'          ','BASE FL ',3*'         ','     STM ','
R/O       ',3*'         ','STM R/O ',3*'         ',9*'        ','     P
REC','IP        ',3*'         ','PRECIP  ',3*'        ','     SUR ',
'R/O       ',3*'         ','SUR R/O ',3*'          ','     INT ','FL
  ',3*'         ','INT FL  ',3*'        ','     BASE',' FL      ',3*'
        ','BASE FL ',3*'          ','     STM ','R/O      ',3*'
  ','STM R/O ',3*'               '/

REAL*8 STSURD(208)      /2*'            ','STORM SU','MMARY  F','ORECAS
T ','RUN  TAB','LC 2    ',13*'          ','WORST   ','NO      ','F/C
AST  ','DIFF    ','%DIFF   ',4*'          ','CASE    ','PRECIP  ',14
*'      ','PEAK    (','CFS)      ',7*'        ',9*'       ','
'PEAK    (','HR)     ',16*'           ','R/O     (','IN)      ',16*'
    ','PRECIP (','IN)     ',7*'      '/

REAL*8 MSVARD(441)/'          ','        MOIS','TURE STO','RAGE VAL',
'UES FORE','CAST RUN','   TABLE',' 3A     ',10*'          ',
'          ','WORST   ','NO      ','F/CAST  ','          ',
'WORST   ','NO      ','F/CAST  ',2*'          ','CASE    ',
'PRECIP  ',2*'          ','CASE    ','PRECIP  ',2*'          ',
9*'        ','         IFS ',3*'      ','         IFS ',4*'
'     UZS  ',3*'      ','     UZS  ',4*'      ',
'     LZS  ',3*'      ','     LZS  ',4*'      ',
'     GWS  ',3*'      ','     GWS  ',4*'      ',
'     BFNX',3*'      ','     BFNX',4*'      ',
'     SIAM',3*'      ','     SIAM',4*'      ',
'     UZC  ',3*'      ','     UZC  ',4*'      ',
9*'
'     IFS  ',3*'      ','     IFS  ',4*'      ',
'     UZS  ',3*'      ','     UZS  ',4*'      ',
'     LZS  ',3*'      ','     LZS  ',4*'      ',
'     GWS  ',3*'      ','     GWS  ',4*'      ',
'     BFNX',3*'      ','     BFNX',4*'      ',
'     SIAM',3*'      ','     SIAM',4*'      ',
'     UZC  ',3*'      ','     UZC  ',4*'      ',261*'        '/

REAL*8 MSVSUD(135)/
'     IFS  ',3*'      ','     IFS  ',4*'      ',
'     UZS  ',3*'      ','     UZS  ',4*'      ',
'     LZS  ',3*'      ','     LZS  ',4*'      ',
'     GWS  ',3*'      ','     GWS  ',4*'      ',
'     BFNX',3*'      ','     BFNX',4*'      ',
'     SIAM',3*'      ','     SIAM',4*'      ',
'     UZC  ',3*'      ','     UZC  ',4*'      ',
9*'
'     IFS  ',3*'      ','     IFS  ',4*'      ',
'     UZS  ',3*'      ','     UZS  ',4*'      ',
'     LZS  ',3*'      ','     LZS  ',4*'      ',
'     GWS  ',3*'      ','     GWS  ',4*'      ',
'     BFNX',3*'      ','     BFNX',4*'      ',
'     SIAM',3*'      ','     SIAM',4*'      ',
'     UZC  ',3*'      ','     UZC  ',4*'      '/

REAL    BLANK/'    '/
REAL*4 STCHAS(648)
REAL*4 STCSS(394)
REAL*4 MSVARY(648)
REAL*4 STSMRY(416)
REAL*4 MSVSUP(270)
EQUIVALENCE (MSVARD(1),MSVARY(1))
EQUIVALENCE (MSVSUD(1),MSVSUP(1))
EQUIVALENCE (STSURD(1),STSMRY(1))
EQUIVALENCE(STCHAR(1),STCHAS(1))
EQUIVALENCE (STCSP(1),STCSS(1))
```

```
                 ------------------------------------------
                 | TYPE =4                                |
                 | MLN  =72                               |
                 | J    =0                                |
                 | MI   =3                                |
                 | NCM  =28                               |
                 ------------------------------------------
                          |
                   +-------------+
                   + DO LOOP TO  +
      ++++++++++++ STMT #     23 +
      +            + K=1,394     +
      +            +-------------+
      +                   |
      +          ------------------------------------------
      +          | STCHAS(110+K) =STCSS(K)                |
      +          ------------------------------------------
      +                   |
23    ++++++++CONTINUE
                          |
                    * * * * *                 ------------------------
                 *     IF      *    T         |                      |
              *   CONOPT(1        * +++++     |     GO TO 61         |
              *5).NE.2            *           |                      |
                 *             *              ------------------------
                    * * * * *
                      F   |
                 ------------------------------------------
                 | J    =3                                |
                 | MI   =1                                |
                 ------------------------------------------
                          |
                   +-------------+
                   + DO LOOP TO  +
      ++++++++++++ STMT #     65 +
      +            + I=1,36      +
      +            +-------------+
      +                   |
      +          ------------------------------------------
      +          | STCHAS(36+I) =SCPAST(I)                |
      +          ------------------------------------------
      +                   |
65    ++++++++CONTINUE
                          |
                 ------------------------------------------
                 | STCHAR(5) =SUBT(3)                     |
                 ------------------------------------------
                          |
                 ------------------------------------------
61               | MAX  =MAXI                             |
                 | STCHAS(55) =MSBDIC                     |
                 | LI   =1                                |
                 ------------------------------------------
                          |
13               CONTINUE
                          |
                 ------------------------------------------
                 | K    =0                                |
                 | L    =0                                |
                 ------------------------------------------
                          |
                   +-------------+
                   + DO LOOP TO  +
      ++++++++++++ STMT #     10 +
      +            + I=1,MAX     +
      +            +-------------+
      +                   |
      +          ------------------------------------------
      +          | IR   =1                                |
      +          | NUM  =2                                |
      +          ------------------------------------------
      +                   |
      +          CALL CONVER(MPDAY(LI),STCHAS(91+K+L),IR,NUM)
      +
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
 +           +-----------+
 +           + DO LOOP TO +
 ++++++++++++ STMT #   11 +
 +           + M=1,MI     +
 +           +-----------+
 +                 |
 +      ---------------------------------------
 +      | NUM  =3                              |
 +      | IR   =0                              |
 +      ---------------------------------------
 +                 |
 +      CALL CONVER(TMPREC(I,M),STCHAS((92+(2*M))+K+L),IR,NUM)
 +                 |
 +            IF            T    CALL CONVER(TMPREC(I,1)
 +        CONOPT(1         +++++ ,STCHAS(97+K+L),IR,
 +      *5).EQ.2
 +               F |+++++++++++++++++++++++++++++
 +      NUM)
 +      CALL CONVER(TMOF(I,M),STCHAS((110+(2*M))+J+K+L),IR,NUM)
 +      CALL CONVER(TMIF(I,M),STCHAS((128+(2*M))+J+K+L),IR,NUM)
 +      CALL CONVER(TMBF(I,M),STCHAS((146+(2*M))+J+K+L),IR,NUM)
 +      CALL CONVER(STMROS(I,M),STCHAS((164+(2*M))+K+L),IR,NUM)
 +                 |
 +            IF            T    CALL CONVER(QOUT(I),STC
 +        CONOPT(1         +++++ HAS(166+K+L),IR,NUM)
 +      *5).EQ.2
 +               F |+++++++++++++++++++++++++++++
 +                 |
 +            IF            T    CALL CONVER(STMROS(I,1)
 +        CONOPT(1         +++++ ,STCHAS(169+K+L),IR,
 +      *5).EQ.2
 +               F |+++++++++++++++++++++++++++++
 +      NUM)
11 ++++++++CONTINUE
 +                 |
 +      ---------------------------------------
 +      | K    =K+108                          |
 +      | LI   =LI+1                           |
 +      ---------------------------------------
 +                 |
 +            IF            T    L=9
 +        MOD(I,4)         +++++
 +      *.EQ.0
 +               F |+++++++++++++++++++++++++++++
 +                 |
 +            IF            T    K=0
 +        MOD(I,4)         +++++
 +      *.EQ.0
 +               F |+++++++++++++++++++++++++++++
 +                 |
 +            IF            T    GO TO 17
 +        MOD(I,8)         +++++
 +      *.EQ.0
 +               F |
```

ORIGINAL PAGE IS
OF POOR QUALITY

C-4

```
                           |
                         . * .
                    * . *  IF    * .  T      ---------------------------------
                    * I.EQ.MAX   *   *****|       GO TO 17                |
                         * . *                ---------------------------------
                         F |
                         -----
                         |10  |
                         -----

17       CALL DVAR(TYPE,STCHAS,MLN,NOM,STRMT,LVAL,HVAL,VOL,TDATA,NCODE)
         ---------------------------------------------
         | STCHAR(7) =SUBT(8)                        |
         | STRMT(4) =SUBT(8)                         |
         ---------------------------------------------
                           |
                         . * .
                    * . *  IF    * .  T      ---------------------------------
                    * I.EQ.MAX   *   *****|       GO TO 10                |
                         * . *                ---------------------------------
                         F |
                   +-----------------+
                   +  DO LOOP TO     +
      ++++++++++++ STMT #     15     +
                   + N=1,325,108     +
                   +-----------------+
                           |
         ---------------------------------------------
         | STCHAS(90+N)  =BLANK                      |
         | STCHAS(99+N)  =BLANK                      |
         ---------------------------------------------
                           |
                   +-----------------+
                   +  DO LOOP TO     +
      ++++++++++++ STMT #     18     +
                   + K=1,6           +
                   +-----------------+
                           |
         STCHAS(92+N+K)=BLANK
         STCHAS(101+N+K)=BLANK
         STCHAS(110+N+K)=BLANK
         STCHAS(119+N+K)=BLANK
         STCHAS(128+N+K)=BLANK
         STCHAS(137+N+K)=BLANK
         STCHAS(146+N+K)=BLANK
         STCHAS(155+N+K)=BLANK
         STCHAS(164+N+K)=BLANK
         STCHAS(173+N+K)=BLANK
18    ++++++++CONTINUE
15    ++++++++CONTINUE
                           |
         ---------------------------------------------
         | K     =0                                  |
         | L     =0                                  |
         ---------------------------------------------
                           |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      +
      +
10    ++++++++CONTINUE
                         IF                       T     ----------------------------------
                      CONOPT(1                    +++++ |     GO TO 32                   |
                    5).NE.2                             ----------------------------------
                         F

           ----------------------------------------------
           | STSURD(5)  =SUBT(3)                        |
           | STSURD(4)  =SUBT(11)                       |
           ----------------------------------------------

               +------------+
               + DO LOOP TO +
      +++++++++ STMT #   39 +
      +        + I=1,36     +
      +        +------------+
      +
      +    ----------------------------------------------
      +    | STSMRY(36+I)  =SSPAST(I)                   |
      +    ----------------------------------------------
      +
39    ++++++++CONTINUE
           ----------------------------------------------
           | IR    =0                                   |
           | NUM   =4                                   |
           ----------------------------------------------

           CALL CONVER(QMAX,STSMRY(95),IR,NUM)
           CALL CONVER(PEAKS(1),STSMRY(98),IR,NUM)
           CALL CONVER(APREC(1),STSMRY(203),IR,NUM)
           CALL CONVER(APREC(1),STSMRY(207),IR,NUM)

           ----------------------------------------------
           | IR    =1                                   |
           | NUM   =3                                   |
           ----------------------------------------------

           CALL CONVER(PHRO,STSMRY(131),IR,NUM)
           CALL CONVER(PHRS(1),STSMRY(134),IR,NUM)

           ----------------------------------------------
           | IR    =0                                   |
           ----------------------------------------------

           CALL CONVER(SUM(1),STSMRY(167),IR,NUM)
           CALL CONVER(SUM(2),STSMRY(170),IR,NUM)

                |35  |

               +------------+
32             + DO LOOP TO +
      +++++++++ STMT #   31 +
      +        + I=1,3      +
      +        +------------+
      +
      +    ----------------------------------------------
      +    | IR    =0                                   |
      +    | NUM   =4                                   |
      +    ----------------------------------------------
      +
      +    CALL CONVER(PEAKS(I),STSMRY(93+(2*I)),IR,NUM)
      +    CALL CONVER(APREC(I),STSMRY(201+(2*I)),IR,NUM)
      +
```

```
       +        -------------------------------------------------
       +        | NUM   =3                                      |
       +        -------------------------------------------------
       +                         |
       +        CALL CCNVER (SUM(I),STSMRY(165+(2*I)),IR,NUM)
       +                         |
       +        -------------------------------------------------
       +        | IR    =1                                      |
       +        -------------------------------------------------
       +                         |
       +        CALL CCNVER(PHRS(I),STSMRY(129+(2*I)),IR,NUM)
       +
 31    ++++++++CCNTINUE
                                 |
 35             -------------------------------------------------
                | IR    =0                                      |
                | NUM   =4                                      |
                | NOM   =12                                     |
                -------------------------------------------------
                                 |
                CALL CONVER(DIFFS,STSMRY(101),IR,NUM)
                CALL CCNVER(DIFFP,STSMRY(137),IR,NUM)
                CALL CCNVER(DIFPP,STSMRY(211),IR,NUM)
                                 |
                       . * * .            ----------------------------
                  . *     IF     * .  T  |     STSMRY(211)=BLANK      |
                *    CONOPT(1         *++++|                            |
                *5).EQ.2              *  |                            |
                  * .           . *       ----------------------------
                       * . *                                   +
                        F |++++++++++++++++++++++++++++++++++++++++
                CALL CCNVER(DIFFMP,STSMRY(103),IR,NUM)
                CALL CCNVER(DIFFPP,STSMRY(139),IR,NUM)
                CALL CCNVER(DIFFR,STSMRY(173),IR,NUM)
                CALL CCNVER(DIFFRP,STSMRY(175),IR,NUM)
                CALL CCNVER(DIFFPR,STSMRY(209),IR,NUM)
                                 |
                       . * * .            ----------------------------
                  . *     IF     * .  T  |     STSMRY(209)=BLANK      |
                *    CONOPT(1         *++++|                            |
                *5).EQ.2              *  |                            |
                  * .           . *       ----------------------------
                       * . *                                   +
                        F |++++++++++++++++++++++++++++++++++++++++
                CALL DVAR(TYPE,STSMRY,MLN,NOM,STST,LVAL,HVAL,TDATA,NCODE)
                                 |
                    +-----------------+
                    + DO LOOP TO      +
       +++++++++++++ STMT #      49   +
       +            + I=1,270         +
       +            +-----------------+
       +                         |
       +        -------------------------------------------------
       +        | MSVARY(378+I) =MSVSUP(I)                      |
       +        -------------------------------------------------
       +                         |
 49    ++++++++CONTINUE
                                 |
                -------------------------------------------------
                | MI    =3                                      |
                -------------------------------------------------
                                 |
                       . * * .            ----------------------------
                  . *     IF     * .  T  |     GO TO 63               |
                *    CONOPT(1         *++++|                            |
                *5).NE.2              *  |                            |
                  * .           . *       ----------------------------
                       * . *
                        F |
```

```
                          |
                    +-----------+
                    + DO LOOP TO +
    +++++++++++++++ STMT #   81 +
    +               + I=1,36     +
    +               +-----------+
    +                     |
    +         ----------------------------------------
    +         | MSVARY(36+I) =MSPAST(I)              |
    +         ----------------------------------------
    +                     |
81  ++++++++CONTINUE
                          |
              ----------------------------------------
              | MSVARD(5) =SUBT(5)                   |
              | MSVARD(6) =SUBT(6)                   |
              | MI    =1                             |
              ----------------------------------------
                          |
              ----------------------------------------
63            | NOM   =40                            |
              | L     =0                             |
              | L1    =1                             |
              | K     =0                             |
              | MAX   =MAXI                          |
              | MSVARY(55) =MSBDIC                   |
              ----------------------------------------
                          |
                    +-----------+
                    + DO LOOP TO +
    +++++++++++++++ STMT #   41 +
    +               + I=1,MAX    +
    +               +-----------+
    +                     |
    +         ----------------------------------------
    +         | IR    =1                             |
    +         | NUM   =2                             |
    +         ----------------------------------------
    +                     |
    +         CALL CONVER(MPDAY(L1),MSVARY(91+K+L),IR,NUM)
    +                     |
    +               +-----------+
    +               + DO LOOP TO +
    +++++++++++++++ STMT #   43 +
    +               + M=1,MI     +
    +               +-----------+
    +                     |
    +         ----------------------------------------
    +         | IR    =0                             |
    +         | NUM   =3                             |
    +         ----------------------------------------
    +                     |
    +         CALL CONVER(EMIFS(I,M),MSVARY(91+(2*M)+K+L),IR,NUM)
    +         CALL CONVER(EMUZS(I,M),MSVARY(109+(2*M)+K+L),IR,NUM)
    +         CALL CONVER(EMLZS(I,M),MSVARY(127+(2*M)+K+L),IR,NUM)
    +         CALL CONVER(EMGWS(I,M),MSVARY(145+(2*M)+K+L),IR,NUM)
    +         CALL CONVER(EMUZC(I,M),MSVARY(199+(2*M)+K+L),IR,NUM)
    +         CALL CONVER(EMSIAM(I,M),MSVARY(181+(2*M)+K+L),IR,NUM)
    +         CALL CONVER(EMBFNX(I,M),MSVARY(163+(2*M)+K+L),IR,NUM)
43  ++++++++CONTINUE
    +                     |
    +         ----------------------------------------
    +         | K     =K+144                         |
    +         | L1    =L1+1                          |
    +         ----------------------------------------
    +                     |
    +                  *  *  *          ------------------------------
    +              *     IF      *    T |       K=0                  |
    +           *    MOD(I,4)      *+++++|                            |
    +           *.EQ.0           *      |                            |
    +              *           *        ------------------------------
    +                  *  *  *                       |
    +                    F |++++++++++++++++++++++++++++
```

```
IF
MOD(I,4)
.EQ.0          T    L=8
F

IF
MOD(I,8)
.EQ.0          T    GO TO 45
F

IF
I.EQ.MAX       T    GO TO 45
F

41

45   CALL DVAR(TYPE,MSVARY,MLN,NOM,MSVT,LVAL,HVAL,VL,TDATA,NCODE)

IF
I.EQ.MAX       T    GO TO 41
F

MSVT(5) =SUBT(2)
MSVARD(8) =SUBT(9)

DO LOOP TO
STMT #     53
N=1,434,144

MSVARY(90+N) =BLANK
MSVARY(98+N) =BLANK

DO LOOP TO
STMT #     55
K=1,7

MSVARY(91+N+K)=BLANK
MSVARY(99+N+K)=BLANK
MSVARY(109+N+K)=BLANK
MSVARY(117+N+K)=BLANK
```

```
    +
    +
    +       MSVARY(127+N+K)=BLANK
    +       MSVARY(135+N+K)=BLANK
    +       MSVARY(145+N+K)=BLANK
    +       MSVARY(153+N+K)=BLANK
    +       MSVARY(163+N+K)=BLANK
    +       MSVARY(171+N+K)=BLANK
    +       MSVARY(181+N+K)=BLANK
    +       MSVARY(189+N+K)=BLANK
    +       MSVARY(199+N+K)=BLANK
    +       MSVARY(207+N+K)=BLANK
    +
 55 ++++++++CCNTINUE
 53 ++++++++CCNTINUE
    +                 |
    +       ----------------------------------------
    +       | K      =0                            |
    +       | L      =0                            |
    +       ----------------------------------------
    +                 |
    +
 41 ++++++++CCNTINUE
            CALL PLTEND(0)
            RETURN
      END
```

```
SUBROUTINE ATTEN

      C    THIS IS THE 2250 OPERATOR INTERFACE ROUTINE

           CCMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
                NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
                IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
                VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
                XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
                YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
                AY     ,BY     ,YLPOS  ,
                LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
                FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
           LOGICAL*1 GSPIN, DUM(2)
           REAL*8 WORK(36)
           INTEGER*4 INTAR(10),GSP1,UNITN,NULLV(1),OAREA
           LOGICAL*1 FA,FB,FC,GMODE1,GSPIN
           LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
           REAL*4 RNTAR(10),XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
           EQUIVALENCE(INTAR(1),RNTAR(1),WORK(1)),(ISYM,ICODE)
           ------------------------------------------
           | ND3  =IABS(IDSP)+1                      |
           ------------------------------------------
           CALL ENATN(IATN,0,28,34)
    5      CONTINUE
           CALL INCL(IGDS5,NULLV,IKEY1)
      C    REQUEST ATTENTION INFORMATION
   10      CALL RQATN(IATN,ICODE,2,INTAR,0,28,34)

                    IF
                  (ICODE)
        -            0            +
    ------       ------       ------
    |20  |       |10  |       |30  |
    ------       ------       ------

   20      CALL THGSP(GSP1)
        +------------------------------------------+
        |  WRITE(NOUT,1000)                        |
        +------------------------------------------+

           STOP
   22      RETURN
      C
   30      CONTINUE

                    IF
                (ICODE-28)
        -            0            +
    ------       ------       ------
    |35  |       |32  |       |35  |
    ------       ------       ------
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                    |
32       | GMODE1 =.NOT.GMODE1                       |
                    |
      +--------------------------------------------------+
     +  WRITE(NOUT,1020)                                +
    +  GMODE1                                          +
   +--------------------------------------------------+
                    |
                 |10  |

35       CONTINUE
                    |
           IF                    T   ---------------------------
         INTAR(1)              *++++|  GO TO 200                |
       .EQ. IGDS6                   ---------------------------
            F
            |
           IF                    T   ---------------------------
         INTAR(1)              *++++|  GO TO 10                 |
       .NE. IGDS5                   ---------------------------
            F
            |
           IF                    T   ---------------------------
         INTAR(2)              *++++|  GO TO 40                 |
       .EQ. IKEY1                   ---------------------------
            F
            |
           IF                    T   ---------------------------
         INTAR(2)              *++++|  GO TO 400                |
       .EQ. IKEY2                   ---------------------------
            F
            |
           IF                    T   ---------------------------
         INTAR(2)              *++++|  GO TO 100                |
       .EQ. IKEY3                   ---------------------------
            F
            |
      C
40       CONTINUE
         CALL DSATN(IATN,0,28,34)
         CALL OMIT(IGDS5,NULLV,IKEY1)
         RETURN
      C
      C  BRING UP PLOT DICTIONARY.  THE DICTIONARY CAN ONLY BE REQUESTED
```

```
      C   FROM THE LATEST DISPLAY
      C
100         CCNTINUE
            CALL OMIT(IGDS5)
            CALL OMIT(IGDSC)
103         CALL INCL(IGDS6)
            ---------------------------------------------
           | ND2   =MINO(ND3,1000)                       |
           | ND1   =1                                    |
            ---------------------------------------------
                          |
                    . * * * .
                 . *   IF     * .
                *  (ND3-1000)   *
              |  * .        . *  |
              |      *  0 |*     |
            - |         |        + |
          -------    -------    -------
          |105  |    |105  |    |104  |
          -------    -------    -------

104         CALL INCL(IGDS6)
            CALL INCL(IGDS6,-3,NULLV)
105         CALL INCL(IGDS6,-2,NULLV)
            CALL INCL(IGDS6,-1,NULLV)
                    +-------------+
       ++++++++++++ + DO LOOP TO  +
       +            + STMT #   110 +
       +            + I=ND1,ND2    +
       +            +-------------+
       +                  |
110    ++++++++++CALL INCL(IGDS6,I,NULLV)
                       -------
                       |110  |
                       -------

      C
      C   PLOT DICTIONARY ATTENTION ROUTINE
      C
200         CCNTINUE
                          |
                    . * * * .
                 . *   IF     * .
                *  (INTAR(4))   *
              |  * .        . *  |
              |      *  0 |*     |
            - |         |        + |
          -------    -------    -------
          |205  |    |205  |    |300  |
          -------    -------    -------
                          |
205                 . * * * .             ---------------------------
                 . *   IF     * .   T    |                           |
                *   INTAR(4)    *+++++   |   GO TO 10                |
                *  .EQ. -1      *        |                           |
                   * .     . *            ---------------------------
                        F |
```

```
                         |
                     . * * .
                  . *   IF     * .   T    --------------------------------
               *   INTAR(4)       *+++++ |  GO TO 250                     |
               * .EQ. -3          *      |                                |
                  * .         . *         --------------------------------
                      *  *
                      F  |
                         |
                     . * * .
                  . *   IF     * .   T    --------------------------------
               *   INTAR(4)       *+++++ |  GO TO 10                      |
               * .NE. -2          *      |                                |
                  * .         . *         --------------------------------
                      *  *
                      F  |
210             CALL OMIT(IGDS6)
                CALL INCL(IGDSC)
                CALL INCL(IGDS5)
                CALL INCL(IGDS5,NULLV,IKEY3)
                CALL INCL(IGDS5,NULLV,IKEY1)
                         |
                     --------
                     |10    |
                     --------

          C
          C   BRING UP PLOT SELECTED IN PLOT DICTIONARY
          C
300             CONTINUE
                         |
                    . * * * .
                 *      IF      * .
               * (INTAR(4)-ND3)    *
           |   * .               . *   |
           |      *  .  *  *  * .      |
           -           0 |             +
     --------       --------       --------
     |310   |       |210   |       |310   |
     --------       --------       --------
310             CONTINUE
                CALL BUFIN(I2250,INTAR(4),IRC)
                CALL INCL(IGDSC)
                         |
                    . * * * .
                 *      IF      * .
               *      (IRC)        *
           |   * .               . *   |
           |      *  .  *  *  * .      |
           -           0 |             +
     --------       --------       --------
     |320   |       |330   |       |320   |
     --------       --------       --------
                         |
320          +-----------------------------------------+
            +  WRITE(NOUT,1010)                       +
           +   IRC                                   +
          +                                         +
         +-----------------------------------------+
                         |
```

```
                  |
               --------
               |10    |
               --------

330          CALL INCL(IGDS5,NULLV,IKEY2)
                  |
               --------
               |10    |
               --------

        C
        C    RETURN FROM A RECALLED DISPLAY
        C
400          CONTINUE
             CALL BUFIN(I2250,ND3,IRC)
                          |
                    * * IF  * *
                 *   (IRC)      *
                    * *    * *
           -             0            +
       --------      --------      --------
       |320   |      |100   |      |320   |
       --------      --------      --------

        C
        C    BRING UP NEXT PAGE OF DICTIONARY
        C
250          CONTINUE
             CALL OMIT(IGDS6)
                          |
                 +----------------+
      +++++++++++ + DO LOOP TO    +
      +           + STMT #   255  +
      +           + I=ND1,ND2     +
      +           +----------------+
      +                   |
255   ++++++++CALL OMIT(IGDS6,I,NULLV)
                          |
                    * * IF  * *            -----------------------
                 *  ND2 .EQ.    *   T      |                     |
                 *  ND3         * +++++    |   GO TO 103         |
                    * *    * *             |                     |
                       F  |                -----------------------
                          |
            ----------------------------------------
            | ND1   =ND2+1                         |
            ----------------------------------------
                          |
                    * * IF  * *
                 *  ((ND3-ND2)-20)  *
                    * *    * *
           -             0            +
       --------      --------      --------
       |265   |      |270   |      |270   |
       --------      --------      --------
```

```
                 |
265     | ND2  #ND3                                  |
        ----------------------------------------------
                 |
              -------
              |104  |
              -------

                 |
        ----------------------------------------------
270     | ND2  =ND2+20                               |
        ----------------------------------------------
                 |
              -------
              |104  |
              -------

    FORMAT 1000 FORMAT(' ATTEN:  PROGRAM TERMINATED BY ''TERMINATE JOB'' OPTION')
    FORMAT 1010 FORMAT(' ***BUFFER WRITE FAILED - R.C. = ',Z8)
    FORMAT 1020 FORMAT(' *GMODE1 CHANGE ',Z2)

END
```

```
SUBROUTINE CHANGE(Y,DPY)
                    |
               REAL * 4 X(92),    Y(366)
         C          KWM STORES DATA IN THE ORDER JAN.(YEAR 2) TO SEPT (YEAR 2)
         C          FOLLOWED BY OCT.(YEAR 1) TO DEC.(YEAR 1).  FEB. 29 IS PLACED
         C          IN DAY 366.
         C          CHANGE REARRANGES THE DATA ARRAY TO STORE THE DATA IN NORMAL
         C          SEQUENCE.  OCT.(YEAR 1) TO  SEPT.(YEAR 2).
                              |
               ------------------------------------------
               | J     =1                               |
               ------------------------------------------
                              |
                    +-----------------+
                    +  DO LOOP TO     +
      +++++++++++++ STMT #      10    +
      +             + I=274,365       +
      +             +-----------------+
      +                       |
      +        ------------------------------------------
      +        | X(J) =Y(I)                             |
      +        ------------------------------------------
      +                       |
      +        ------------------------------------------
   10 +++++++++| J     =J+1                             |
               | K     =365                             |
               | J     =273                             |
               ------------------------------------------
                              |
                          *   .   *                   ----------------------------
                   .  *       IF       *  .   T       |   GO TO 15               |
                 *      DPY  .NE.             *+++++  |                          |
                 * 366                        *       |                          |
                      *  .          .  *              ----------------------------
                           *   .   *
                           F   |
         C          IF LEAP YEAR MOVE FEB. 29TH
                              |
               ------------------------------------------
               | TEMP  =Y(366)                          |
               | K     =366                             |
               ------------------------------------------
                              |
   15          CONTINUE
                              |
                    +-----------------+
                    +  DO LOOP TO     +
      +++++++++++++ STMT #      20    +
      +             + I=1,214         +
      +             +-----------------+
      +                       |
      +        ------------------------------------------
      +        | Y(K) =Y(J)                             |
      +        | K     =K-1                             |
      +        ------------------------------------------
      +                       |
      +        ------------------------------------------
   20 +++++++++| J     =J-1                             |
               ------------------------------------------
                              |
                          *   .   *                   ----------------------------
                   .  *       IF       *  .   T       |   Y(152) = TEMP          |
                 *      DPY  .EQ.             *+++++  |                          |
                 * 366                        *       |                          |
                      *  .          .  *              ----------------------------
                           *   .   *                                   +
                           F   |+++++++++++++++++++++++++++++++++++++++++
```

```
                       +---------------+
                       +  DO LOOP TO   +
           ++++++++++++ STMT #      25 +
           +           + I=1,59        +
           +           +---------------+
           +                   |
           +           ---------------------------------------------
           +           | J     =I+92                               |
           +           ---------------------------------------------
           +                   |
           +           ---------------------------------------------
      25   ++++++++++++| Y(J) =Y(I)                                |
                       ---------------------------------------------
                               |
                       +---------------+
                       +  DO LOOP TO   +
           ++++++++++++ STMT #      30 +
           +           + I=1,92        +
           +           +---------------+
           +                   |
           +           ---------------------------------------------
      30   ++++++++++++| Y(I) =X(I)                                |
                       ---------------------------------------------
                               |
                       RETURN

      END
```

```
SUBROUTINE CONVER(X,A,IR,NUM)

          REAL CON(24)
          DATA CON/100000.,8.,10000.,7.,1000.,6.,100.,5.,10.,4.,1.,3.,.1,
          2.,.01,2.,.001,2.,.0001,2.,.00001,2.,.000001,2./
          INTEGER*4 A(2),B(2),IR,NUM
          DATA B/2*4H    /
          EQUIVALENCE (Z,IX)
          ------------------------------------------
          | Z      =X                              |
          | F      =Z                              |
          | IF(IR.EQ.1)F =IX                       |
          | IS     =64                             |
          ------------------------------------------
                      |
                      IF                   T   IS=96
                   F.LT.0.0            *++++
                    F |++++++++++++++++++++++++++++
                      |
                      IF                   T   F=-F
                   IS.EQ.96            *++++
                    F |++++++++++++++++++++++++++++
          ------------------------------------------
          | M      =2                              |
          ------------------------------------------
                      |
                      IF                   T   M=1
                   (IR.EQ.1            *++++
                 *).AND.(NUM.LE.4)     *
                    F |++++++++++++++++++++++++++++
               +--------------+
               + DO LOOP TO   +
    +++++++++++ STMT #     9  +
    +          + I=1,M        +
    +          +--------------+
    +                 |
    +     ------------------------------------------
    +     | A(I) =B(I)                             |
    +     ------------------------------------------
    +                 |
  9 ++++++++CONTINUE
                      |
          ------------------------------------------
          | L      =1                              |
          ------------------------------------------
                      |
                      IF                   T   GO TO 15
                   (F-10000            *++++
                 *0.).GE.0.            *
                    F |
          ------------------------------------------
          | L      =3                              |
          ------------------------------------------
                      |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
              |
          . * * * .
      . *    IF     * .   T
     *   (F-10000        *+++++ | GO TO 15 |
     *.).GE.0.           *
       * .           . *
           *   .   *
           F   |
    | L      =5                          |
               |
          . * * * .
      . *    IF     * .   T
     *   (F-1000.        *+++++ | GO TO 15 |
     *).GE.0.            *
       * .           . *
           *   .   *
           F   |
    | L      =7                          |
               |
          . * * * .
      . *    IF     * .   T
     *   (F-100.)        *+++++ | GO TO 15 |
     *.GE.0.             *
       * .           . *
           *   .   *
           F   |
    | L      =9                          |
               |
          . * * * .
      . *    IF     * .   T
     *   (F-10.).        *+++++ | GO TO 15 |
     *GE.0.              *
       * .           . *
           *   .   *
           F   |
    | L      =11                         |
               |
          . * * * .
      . *    IF     * .   T
     *   (F-1.).G        *+++++ | GO TO 15 |
     *E.0.               *
       * .           . *
           *   .   *
           F   |
               |
          . * * * .
      . *    IF     * .   T
     *   ((F-0.)         *+++++ | GO TO 23 |
     *.EQ.0.) .AND. (IR .*
     *EQ.1)              *
       * .           . *
           *   .   *
           F   |
    | L      =13                         |
               |
 15 | ICNT  =CON(L+1)                    |
               .
```

```
                CALL ISCHAR(IS,1,A)
                     |
                -----------------------------------------
                | NUMA  =NUM+2                          |
                -----------------------------------------
                     |
             +------------+
             + DO LOOP TO +
  ++++++++++ + STMT #   10 +
  +          + I=2,NUMA   +
  +          +------------+
  +                  |
  +             -----------------------------------------
  +             | J2    =0                              |
  +             | J1    =F/CON(L)                       |
  +             | F     =F-J1*CCN(L)                    |
  +             | J1    =J1+240                         |
  +             -----------------------------------------
  +                  |
  +            . * * .                  ---------------------------
  +         . *   IF    * .  T          |                         |
  +       *  ((ICNT-I        *++++      |   GO TO 12              |
  +       *).EQ.0).AND.(IR.EQ.*         |                         |
  +       *1)                 *         |                         |
  +         * .           . *           ---------------------------
  +            * . *
  +            F |
  +              |
  +            . * * .                  ---------------------------
  +         . *   IF    * .  T          |                         |
  +       *  (ICNT-I)        *++++      |   J2=75                 |
  +       *.EQ.0              *         |                         |
  +         * .           . *           ---------------------------
  +            * . *                                             +
  +            F |++++++++++++++++++++++++++++++++++++++++++++++++
  +            . * * .                  ---------------------------
  +         . *   IF    * .  T          |                         |
  +       *  J2.EQ.75        *++++      |   CALL ISCHAR(J2,I,A)   |
  +         * .           . *           ---------------------------
  +            * . *                                             +
  +            F |++++++++++++++++++++++++++++++++++++++++++++++++
  +            . * * .                  ---------------------------
  +         . *   IF    * .  T          |                         |
  +       *  J2.EQ.75        *++++      |   I=I+1                 |
  +         * .           . *           ---------------------------
  +            * . *                                             +
  +            F |++++++++++++++++++++++++++++++++++++++++++++++++
  +             CALL ISCHAR(J1,I,A)
  +                  |
  +             -----------------------------------------
  +             | L     =L+2                            |
  +             -----------------------------------------
  +                  |
10 ++++++++CCNTINUE
                     |
                 ---------
                 |12     |
                 ---------

                     |
                -----------------------------------------
23              || J1   =240                            |
                -----------------------------------------
                     |
                CALL ISCHAR(J1,1,A)
12              RETURN
END
```

```
BROUTINE CORREL IS USED TO CALCULATE CORRELATION COEFFICIENT
      C       X1 =  ARRAY 1   ,  X2 = ARRAY2 , IV = NO.OF VARIABLES IN ARRAY
      C     ANS =  ANSWER OF CORRELATION COEFFICIENT
      C
            SUBROUTINE CORREL (X1, X2, IV, ANS)
      C
            DIMENSION  X1(1), X2(1)
            REAL*4  X1, X2, TX1, TX2, MX1, MX2, DX1, DX2, AX1, AX2, AX3, ANS
      C
      C
            ------------------------------------------------
            | AX1  =0.0                                    |
            | AX2  =0.0                                    |
            | AX3  =0.0                                    |
            | DX1  =0.0                                    |
            | DX2  =0.0                                    |
            | MX1  =0.0                                    |
            | MX2  =0.0                                    |
            | TX1  =0.0                                    |
            | TX2  =0.0                                    |
            ------------------------------------------------
      C
      C  DETERMINE THE MEAN OF ARRAY X1, X2
      C
                  +-----------+
                  + DO LOOP TO +
      ++++++++++++ STMT #   100 +
      +           + I = 1,IV   +
      +           +-----------+
      +
      +     ------------------------------------------------
      +     | TX1  =TX1+X1(I)                              |
      +     ------------------------------------------------
      +
      +     ------------------------------------------------
  100 ++++++++| TX2  =TX2+X2(I)                              |
            | MX1  =TX1/IV                                 |
            | MX2  =TX2/IV                                 |
            ------------------------------------------------
      C
                  +-----------+
                  + DO LOOP TO +
      ++++++++++++ STMT #   101 +
      +           + I = 1,IV   +
      +           +-----------+
      +     ------------------------------------------------
      +     | DX1  =X1(I)-MX1                              |
      +     ------------------------------------------------
      +
      +     ------------------------------------------------
      +     | DX2  =X2(I)-MX2                              |
      +     | AX1  =AX1+DX1*DX1                            |
      +     | AX2  =AX2+DX2*DX2                            |
      +     ------------------------------------------------
      +
      +     ------------------------------------------------
  101 ++++++++| AX3  =AX3+DX1*DX2                              |
            | ANS  =AX3/SQRT(AX1*AX2)                      |
            ------------------------------------------------
            RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
SUBROUTINE DAYNXT(DAY,DPY)

        C  DETERMINES NUMBER OF NEXT DAY OF THE YEAR

            INTEGER   DAY,DPY
            ----------------------------------------
            | DAY  =DAY+1                          |
            ----------------------------------------
                       |
                       .                --------------------------
                  . *     * .           |                        |
            . *     IF      * .    T    |    DAY = 1             |
         *      DAY .EQ.       *++++++  |                        |
         * 366                 *        |                        |
            *  .           .  *         --------------------------
                  *  .  *
                 F |++++++++++++++++++++++++++++++++
                       .                --------------------------
                  . *     * .           |                        |
            . *     IF      * .    T    |    DAY = 366           |
         *.     DAY .EQ.       *++++++  |                        |
         * 60 .AND. DPY  .EQ.  *        |                        |
         *366                  *        |                        |
            *  .           .  *         --------------------------
                  *  .  *
                 F |++++++++++++++++++++++++++++++++
                       .                --------------------------
                  . *     * .           |                        |
            . *     IF      * .    T    |    DAY = 60            |
         *      DAY .EQ.       *++++++  |                        |
         * 367                 *        |                        |
            *  .           .  *         --------------------------
                  *  .  *
                 F |++++++++++++++++++++++++++++++++

            RETURN

END
```

```
SUBROUTINE DAYOLT(VDCY,MEDWY,DPY)

        C  PRINTS TABLE OF DAILY VALUES

           DIMENSION  MEDWY(12),VDCY(366),VDMD(12)
           INTEGER  DATE,DAY,DPY

100        WRITE(6,1)

                              FORMAT    1 FORMAT (7X,'DAY',7X,'OCT'
                              ,5X,'NOV',5X,'DEC',5X,'JAN',5X,'FEB',
           5X,'MAR',5X,'APR',5X,'MAY',5X,'JUNE',4X,'JULY',4X,'AUG',5X,
           'SEPT',8X,'ANNUAL')

           MEDWY(3) =0

           DO LOOP TO
           STMT #    104
           DATE = 1,28,1

           IF
           MOD(DATE
           ,5) .NE. 1       T  ---->  GO TO 102
           F

           DO LOOP TO
           STMT #    101
           KMO = 1,12

           DAY  =MEDWY(KMO)+DATE

101        VDMD(KMO) =VDCY(DAY)

           WRITE(6,2)
           DATE,VDMD(12),(VDMD(KWD), KWD=1,11)

                              FORMAT    2 FORMAT(1H0,3X,I6,3X,12F8.1)

           104

102        DO LOOP TO
           STMT #    103
           KMO = 1,12

           DAY  =MEDWY(KMO)+DATE
```

```
103 ++++++++| VDMD(KMO) =VDCY(DAY)                            |

            WRITE(6,3)
            DATE,VDMD(12),(VDMD(KWD), KWD = 1,11)

                                        FORMAT     3 FORMAT(1X,3X,I6,3X,12F8.1)

104 ++++++++CONTINUE

               IF                T
          DPY .NE.                +++++|    GO TO 106        |
       366

               F

        | DATE =29
        | TEMP =VDCY(60)
        | VDCY(60) =VDCY(366)                                  |

              DO LOOP TO
    ++++++++++ STMT #     105
              KMO = 1,12

        | DAY   =MEDWY(KMO)+DATE                               |

105 ++++++++| VDMD(KMO) =VDCY(DAY)                             |

            WRITE(6,3)
            DATE,VDMD(12),(VDMD(KWD), KWD=1,11)

                 |107  |

106         CONTINUE

            WRITE(6,4)
            VDCY(302),VDCY(333),VDCY(363),VDCY(29),VDCY(88)+

            VDCY(119),VDCY(149),VDCY(180),VDCY(210),VDCY(241),VDCY(272)
                                        FORMAT     4 FORMAT(1X,7X,2H29,3X
                                                   ,4F8.1,8X,7F8.1)
107         CONTINUE
108
            WRITE(6,5)
            VDCY(303),VDCY(334),VDCY(364),VDCY(30),VDCY(89)+
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
        VCCY(120),VDCY(150),VDCY(181),VDCY(211),VDCY(242),VDCY(273)
                                         FORMAT     5 FORMAT(1X,7X,2H30,3X,
                  |                                 4F8.1,8X,7F8.1)
       +---------------------------------------------------+
      +  WRITE(6,6)                                        +
     +   VDCY(304),VDCY(365),VDCY(31),VDCY(90),VDCY(151)+
    +                                                   +
   +                                                   +
  +-------------------------------------------------+
                  |
        VDCY(212),VDCY(243)
                                         FORMAT     6 FORMAT(1H/,7X,2H31,3X
                  |                  ,F8.1,8X,2F8.1,8X,F8.1,6X,F8.1,8X,2F8.1)
        ---------------------------------------
        | MEDWY(3) =365                       |
        ---------------------------------------
                  |
                  .                          ---------------------------
             . *  *  * .                     |                         |
       * . *  DPY  IF     * .  T             |   VDCY(60) = TEMP       |
       * 366       .EQ.         *++++        |                         |
         *  .      *  *   . *                |                         |
                   .                         ---------------------------
                   F                                    +
                  |+++++++++++++++++++++++++++++++++++++++
        RETURN

  END
```

```
SUBROUTINE DICTRY(TEXT)

      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     NULLV  ,IGDSC  ,IPINQW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     AY     ,BY     ,YLPOS  ,
     LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      LOGICAL*1 GSPIN, DUM(2)
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      LOGICAL*1 FA,FB,FC,GMODE1,GSPIN
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LOGICAL*1 LINE(49)/47*' ','@',' '/
      LOGICAL*1 TEXT(72)
      REAL*8 WORK(36)
      EQUIVALENCE(RWORK1,IABSDP)
C
C   CREATE DICTIONARY ENTRY
C
                   |
      ---------------------------------------------
      | IABSDP =IABS(IDSP)+1                      |
      ---------------------------------------------
                   |
                 * * *
            *      IF       *
           *     (IDSP)      *
       |    *               *    |
       |         * * *           |
     - |           0 |         + |
   --------    ---------    ---------
   |202   |    |205    |    |205    |
   --------    ---------    ---------
                   |
202   ---------------------------------------------
      | LINE(1)  =TEXT(1)                         |
      | LINE(2)  =TEXT(2)                         |
      | LINE(3)  =TEXT(3)                         |
      ---------------------------------------------
                   |
           +-----------------+
           + DO LOOP TO      +
 ++++++++++ STMT #     203   +
 +         + I=4,47          +
 +         +-----------------+
 +                 |
 +    ---------------------------------------------
203 ++++++| LINE(I) =LINE(49)                     |
      ---------------------------------------------
                   |
               ---------
               |215    |
               ---------

                   |
205        +-----------------+
           + DO LOOP TO      +
 ++++++++++ STMT #     210   +
 +         + I=1,47          +
 +         +-----------------+
 +                 |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
              +                 |
              +                 |
210 ++++++++| LINE(I) =TEXT(I)                                  |
              ---------------------------------------------------
                                |
215           CALL PTEXT(IGDS6,LINE,48,IABSDP,NULLV,2,0.,YLPOS)
              CALL EXEC(IGDS6)
              ---------------------------------------------------
              | YLPOS=YLPOS-1.5*VSPL                            |
              ---------------------------------------------------
                                |
                                .              ---------------------------------------------
                        . *  *  *   *          |                                           |
                     . *      IF      * .  T   |    YLPOS=RU-2.5*VSPL                      |
                    *      YLPOS .L         *+++++                                          |
                    *T. 376.                *  |                                           |
                       * .            . *      |                                           |
                            *    *             ---------------------------------------------
                            F   |                             +
                                |++++++++++++++++++++++++++++++
              RETURN
          END
```

```
SUBROUTINE DLPO(IPAR,MLN,NOM,OPAR,NCODE)
          INTEGER *4 TYPE/1/,LVAL(1)/1/,HVAL(1)/1/,VDL/1/,TDATA/1/
          INTEGER *4 IPAR(192)
          INTEGER *4 OPAR(32)
          CALL DVAR(TYPE,IPAR,MLN,NOM,OPAR,LVAL,HVAL,VDL,TDATA,NCODE)
          RETURN
END
```

```
SUBROUTINE DMSG(IPAR,MLN,NOM,NCODE)
          INTEGER *4 TYPE/4/,LVAL(1)/1/,HVAL(1)/1/,VDL/1/,TDATA/1/
          INTEGER *4 IPAR(192)
          INTEGER *4 OPAR(32)
          CALL DVAR(TYPE,IPAR,MLN,NOM,OPAR,LVAL,HVAL,VDL,TDATA,NCODE)
          RETURN
END
```

```
C*     TITLE:   DISPLAY / OPTION PROCESSING MODULE (DVAR)                    *
C*                                                                           *
C*     FUNCTION: DVAR ALLOWS THE CALLER TO COMMUNICATE WITH THE              *
C*               USER(S) VIA THE 2250 DISPLAY CONSOLE TERMINAL.              *
C*              DVAR PROCESSES LIGHT PEN OPTIONS, VARIABLE DATA              *
C*               OPTION(S), AND TUTORIAL TYPE MESSAGE(S), AND                *
C*               DATA SET UPDATING.                                          *
C*               THE USER(S) RESPONSE(S), LIGHT PEN(S) OR KEYBOARD           *
C*               ACTION(S), AS APPLICABLE, ARE RETURNED TO THE               *
C*               CALLING PROGRAM.                                            *
C*                                                                           *
C*     ENTRY POINT:  DVAR   CALLED BY USING PROGRAM                          *
C*     EXIT: RETURN TO CALLING PROGRAM                                       *
C*                                                                           *
C*     PARM LIST AND DEFINITIONS:                                            *
C*        1. TYPE  -  TYPE OF REQUEST - 1 = LP ACTION, 2 = VAR. DATA         *
C*                                      3 = DATA SET UPDATING                *
C*                                      4 = MESSAGES                         *
C*        2. IPAR  -  LIST OF OPTION(S), MESSAGES, OR DATA SET               *
C*                         RECORDS TO BE TO BE DISPLAYED AND/OR UPDATED      *
C*        3. MLN  -  SIZE IN BYTES OF EACH MEMBER IN IPAR ARRAY              *
C*        4. NUM  -  NO. OF IPAR MEMBERS                                     *
C*        5. OPAR  -  AN ARRAY USED TO RETURN USER(S) SELECTIONS             *
C*        6. LVAL  -  LO VALUE ARRAY (VARIABLE DATA)                         *
C*        7. HVAL  -  HI VALUE ARRAY (VARIABLE DATA)                         *
C*        8. VDL  -  MAX. CHARS. USER CAN ENTER FROM 2250 KEYBOARD           *
C*        9. TDATA -  LEGAL VAR. DATA - FLT. PT. = 1,  INTG. = 2             *
C*       10. NCODE - RETURN CODE                                             *
C*                                                                           *
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE DVAR(TYPE,IPAR,MLN,NOM,OPAR,LVAL,HVAL,VDL,TDATA,NCODE)
      CCMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     *     NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     *     IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     *     VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     *     XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     *     YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     *     AY     ,BY     ,YLPOS  ,
     *     LOGXSW,LGGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     *   FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL,IFLAG
      LOGICAL*1 GSPIN, DUM(2)
      INTEGER*4 GSP1,UNITN,OAREA,NULLV(1),STAR(19)
      REAL*8 IDF(9),WORK(36)
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6),GLAB(16)
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LCGICAL*1 FA,FB,FC,GMODE1
      EQUIVALENCE (WORK(1),STAR(1)),(WCRK(11),GLAB(1))
      EQUIVALENCE  (NULL,NULLV)
      COMPLEX *16 HDR(21)/'SELECT OPTION(S)',' OR CONTINUE    ',
     *                   'SELECT VARIABLE ','TO UPDATE       ',
     *                  'SELECT LINE TO B','E UPDATED       ',
     *                   'SELECT CONTINUE ',
     *                   'TO RESUME       ',
     *                   '****  INVALID - ','ENTER AGAIN  ***',
     *                   '*
     *                   'ENTER FLT. PT. O','R INTEGER AS APP',
     *                   'LICABLE         ',
     *                   '                ','                ',
     *                   '                '/
      COMPLEX *16 BLANKS(5)/'                ','                ',
     *                      '                ','                ',
     *                      '                '/
      CCMPLEX *16  BLINE(5)/'||||||||||||||||','||||||||||||||||',
     *                      '||||||||||||||||','||||||||||||||||',
     *                      '||||||||||||||||'/
      COMPLEX *16   RBUF(5)/'                ','                ',
     *                      '                ','                ',
     *                      '                '/
      INTEGER*4 PREFIX(32)/' 1. ',' 2. ',' 3. ',' 4. ',
     *                     ' 5. ',' 6. ',' 7. ',' 8. ',
     *                     ' 9. ','10. ','11. ','12. ',
     *                     '13. ','14. ','15. ','16. ',
     *                     '17. ','18. ','19. ','20. ','21. ','22. ',
     *                     '23. ','24. ','25. ','26. ',
     *                     '27. ','28. ','29. ','30. ','31. ','32. '/
      COMPLEX *16 SUFFIX/'----------------'/
      COMPLEX *16  FOOT(2)/'        CONTINUE','----@           '/
      COMPLEX *16 FOOT1(2)/'        TERMINAT','E---@           '/
      COMPLEX *16 FOOT2(2)/'        BACKWARD','----@           '/
      INTEGER *2 DISP/1/
      INTEGER *4 ECODE
      INTEGER*4 IPAR(1)
      INTEGER *2 LP/'@ '/
      INTEGER *4 LPCA(10)
      INTEGER *4 MSG(7)
      INTEGER *4 TYDATA
      INTEGER *4 TYPE
```

```
      INTEGER *4 TDATA
      INTEGER *4 VDL
      INTEGER *4 Y
      INTEGER *4 Z3,Z4
      REAL *4 HVAL(32)
      REAL *4 LVAL(32)
      REAL*4    DPAR(1)
      EQUIVALENCE(IALEV,IATN)
      EQUIVALENCE (MIDSP,IDSP)
      REAL *8 VDATA/'||||||||'/
      REAL *4 Z1,Z2
      EQUIVALENCE (X,Y)
      EQUIVALENCE   (Z1,Z3),(Z2,Z4)
C*
C**  USER'S OPTION(S) PROCESSING
C*
10000 CONTINUE
                                             FORMAT    1 FORMAT('0 DVAR DISLAY
                                            ACTION - TYPE =',I4,' TDATA =',I4)
          IF
       IFLAG.EQ.1               T           GO TO 11001
          F
C   INITIALIZE GRAPHIC SERVICES
      CALL INGSP(GSP1,NULLV)
C   SPECIFY SUBROUTINE LINK/LOAD STATUS
      CALL SPEC(GSP1,2,17,19,26,-28,53,-56)
      CALL SPEC(GSP1,1,7,8,16,29,-31,37,-40)
C   INITIALIZE THE 2250
      CALL INDEV(GSP1,UNITN,I2250)
C   INITIALIZE GRAPHIC DATA SETS
      CALL INGDS(I2250,IGDSC,3072)
      CALL INGDS(I2250,IGDS5,128)
      CALL INGDS(I2250,IGDS6,512)
C   SET NO SCISSIORING FOR ALL DATA SETS
      CALL SSCIS(IGDSC,3)
      CALL SSCIS(IGDS5,3)
      CALL SSCIS(IGDS6,3)
C*
C**  SET UP TO BUILD DATA SET(S)
```

```
      C*
11001      CONTINUE
                 |
             . * . * .
          . *    IF     * .   T  ------------------------------------
        *  TYPE  .EQ      *****|        GO TO 2000                  |
        *. 4               *   |                                    |
          *  .          .  *   ------------------------------------
             *  .  *
              F |
           CALL INGDS(I2250,IGDS,1536)
           CALL INGDS(I2250,MSG(1),128,NULL,MSG(2),MSG(3),MSG(4),MSG(5),
                     MSG(6),MSG(7))
           CALL SGRAM(IGDS,2)
           CALL SDATM(IGDS,3)
           CALL CRATL(I2250,IALEV)
           CALL SLPAT(IGDS,1)
                 |
           ----------------------------------------------
           | NCHMOD =4                                  |
           | NX    =48                                  |
           | NY    =20                                  |
           ----------------------------------------------
                 |
             . * . * .
          . *    IF     * .   T  ------------------------------------
        *  TYPE.NE.        *****|        GO TO 11010                 |
        *3                 *   |                                    |
          *  .          .  *   ------------------------------------
             *  .  *
              F |
           ----------------------------------------------
           | NCHMOD =3                                  |
           | NX    =72                                  |
           | NY    =49                                  |
           ----------------------------------------------
                 |
11010      CONTINUE
           CALL SCHAM(IGDS,NCHMOD)
           CALL SDATL(IGDS,0,0,NX,NY)
      C*
      C**  BUILD MSG DATA SET(S )
      C*
                 |
             +-------------+
             + DO LOOP TO  +
  +++++++++++ STMT # 11002 +
  +          + N=1,7       +
  +          +-------------+
  +              |
  +        CALL SCHAM(MSG(N),NCHMOD-2)
  +        CALL SGRAM(MSG(N),2)
  +        CALL SDATM(MSG(N),3)
  +        CALL SGDSL(MSG(N),0,NY-1,NX,NY,0,0,NX,NY)
  +        CALL PTEXT(MSG(N),HDR(3*N-2),48,1,NULL,1,0,NY)
  +
11002 ++++++++CONTINUE
      C*
      C**  SET UP TO COMMUNICATE WITH 2250 OPERATOR
```

```
        C*

11003          CCNTINUE
                  |
              +-------------+
              + DO LOOP TO  +
   +++++++++++ STMT # 11004 +
   +          + N=1,6       +
   +          +-------------+
   +                |
   +          CALL INCL(MSG(N))
   +
11004 ++++++++CCNTINUE
                  |
               CALL PTEXT(IGDS,FOOT ,32,100,NULL,1,0,6)
                  |
                * * *          -------------------------
             *    IF     *  T  |       GO TO 5          |
          *    TYPE.NE.    *++++|                        |
          *3               *    |                        |
             *           *      -------------------------
                * * *
                F |
               CALL PTEXT(IGDS,FOOT1,32,101,NULL,1,0,4)
               CALL PTEXT(IGDS,FOOT2,32,102,NULL,1,0,2)
     5         CCNTINUE
               CALL PTEXT(IGDS,BLANKS,NX,50,NULL,1,0,1)
                  |
               -------------------------------------------
               | ECODE=4                                 |
               -------------------------------------------
                  |
               GO TO (10,10,3000,2000), TYPE
                  |
                 ------
                 |1040 |
                 ------

    10         CCNTINUE
                  |
               -------------------------------------------
               | ECODE=8                                 |
               -------------------------------------------
                  |
                * * *          -------------------------
             *    IF     *  T  |       GO TO 1040       |
          *    MLN.LE.0    *++++|                        |
          *.OR.MLN.GT.12   *    |                        |
             *           *      -------------------------
                * * *
                F |
               -------------------------------------------
               | ECODE=12                                |
               -------------------------------------------
                  |
                * * *          -------------------------
             *    IF     *  T  |       GO TO 1040       |
          *    NOM.LE.0    *++++|                        |
          *.OR.NOM.GT.32   *    |                        |
             *           *      -------------------------
                * * *
                F |
                * * *          -------------------------
             *    IF     *  T  |       GO TO 20         |
          *    TYPE.EQ.    *++++|                        |
          *1               *    |                        |
             *           *      -------------------------
                * * *
                F |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                      |
 -----------------------------------------------
 | ECODE=16                                    |
 -----------------------------------------------
                      |
               . * . * .           ---------------------------------
         . *       IF       * .  T |        GO TO 1040             |
        *      VDL.LE.0         *++++|                               |
        *.CR.VDL.GT.6           *   |                               |
            * .     . *            ---------------------------------
                F  |
 -----------------------------------------------
 | ECODE=20                                    |
 -----------------------------------------------
                      |
               . * . * .           ---------------------------------
         . *       IF       * .  T |        GO TO 1040             |
        *     TDATA.LE          *++++|                               |
        *.0.OR.TDATA.GT.2       *   |                               |
            * .     . *            ---------------------------------
                F  |
C*
C**  SET UP TO ACCESS I/P ARRAY MEMBERS
C*
20          CONTINUE
                      |
 -----------------------------------------------
 | ECODE=0                                     |
 | X     =NOM                                  |
 | X     =(X/2.0)+0.75                         |
 | M     =X                                    |
 -----------------------------------------------
                      |
50          CONTINUE
                      |
 -----------------------------------------------
 | X     =MLN                                  |
 | X     =(X/4.0)+0.75                         |
 | I     =X                                    |
 | LTH   =4*I                                  |
 -----------------------------------------------
                      |
C*
C**  BUILD VARIABLE CR OPTION(S) DATA SET
C*
                      |
 -----------------------------------------------
 | NCV   =0                                    |
 -----------------------------------------------
                      |
              +----------------+
              +  DO LOOP TO    +
 ++++++++++++++  STMT #    100 +
 +            +  L=1,M,1       +
 +            +----------------+
 +                    |
 +  -----------------------------------------------
 +  | NCV   =NCV+1                                |
 +  -----------------------------------------------
 +                    |
 +  CALL PTEXT(IGDS,PREFIX(L),3,NULL,NULL,1,0,19-NCV)
 +  CALL PTEXT(IGDS,IPAR(I*(L-1)+1),LTH,NULL,NULL,1,4,19-NCV)
 +
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
              *
          . * . * * .            T  ---------------------------
      *  .  *    IF        *  .    |     GO TO 99              |
      *       MLN.GE.1           *++++|                           |
      *1                         *   |                           |
          *  .           .  *        ---------------------------
              *  .  *
              F |

      CALL PTEXT(IGDS,SUFFIX,(11-MLN),NULL,NULL,1,MLN+4,19-NCV)
 99 + CONTINUE
      CALL PTEXT(IGDS,LP,2,L,NULL,1,16,19-NCV)
100 ++++++++CONTINUE
              |
          . * . * * .            T  ---------------------------
      *  .  *    IF        *  .    |     GO TO 299             |
      *       M.EQ.1.A           *++++|                           |
      *ND.NOM.EQ.1               *   |                           |
          *  .           .  *        ---------------------------
              *  .  *
              F |

      --------------------------------------------
      | ICV    =NCV                               |
      | M      =M+1                               |
      --------------------------------------------
              |
       +-----------------+
       +  DO LOOP TO     +
++++++++++ STMT #    200 +
 +     +  L=M,NOM,1      +
 +     +-----------------+
 +            |
 +    --------------------------------------------
 +    | NCV    =NCV+1                             |
 +    --------------------------------------------
 +            |
 +    CALL PTEXT(IGDS,PREFIX(L),3,NULL,NULL,1,25,ICV+19-NCV)
 +    CALL PTEXT(IGDS,IPAR(I*(L-1)+1),LTH,NULL,NULL,1,29,ICV+19-NCV)
 +            |
 +        . * . * * .            T  ---------------------------
 +    *  .  *    IF        *  .    |     GO TO 199             |
 +    *       MLN.GE.1           *++++|                           |
 +    *1                         *   |                           |
 +        *  .           .  *        ---------------------------
 +            *  .  *
 +            F |
 +
 +    CALL PTEXT(IGDS,SUFFIX,(11-MLN),NULL,NULL,1,MLN+29,ICV+19-NCV)
199 + CONTINUE
 +    CALL PTEXT(IGDS,LP,2,L,NULL,1,41,ICV+19-NCV)
200 ++++++++CONTINUE
     C*
     C**  DISPLAY MAIN OPTIONS
     C*
299       CONTINUE
          CALL SALRM(I2250)
300       CONTINUE
          CALL EXEC(IGDS)
          CALL EXEC(MSG(TYPE))
          CALL ENATN(IALEV,34)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      C*
      C**  PROCESS LP ACTION
      C*
           CALL RQATN(IALEV,NCODE,2,LPCA,34)
           CALL DSATN(IALEV,34)
                        |
           ----------------------------------------------
           | LPSEL=LPCA(4)                              |
           ----------------------------------------------
                        |
                  . * * .
              . *    IF     * .
            *     (LPSEL)      *
          | *  .             . * |
          |     *  .  * .  *     |
          -       0 |           +
      -------   -------     -------
      |300  |   |300  |     |310  |
      -------   -------     -------

310        CONTINUE
                        |
                  . * * .
              . *    IF      * .  T            ---------------------------
            *     LPSEL.EQ        *++++++      |    GO TO 1050           |
            *.100 .OR. LPSEL.EQ.*              |                         |
            *101  .OR. LPSEL.EQ.1*             |                         |
                *  .       . *                 ---------------------------
                     * F  *
                        |
                  . * * .
              . *    IF      * .  T            ---------------------------
            *      TYPE.EQ.       *++++++      |    GO TO 300            |
            *4.OR.LPSEL.GT.NCM    *            |                         |
                *  .       . *                 ---------------------------
                     * F  *
                        |
320        CONTINUE
                        |
         +--------------------------------------------------+
        +  WRITE (6,330)                                   +
       +    LPSEL                                         +
      +--------------------------------------------------+
                        |
                                      FORMAT   330 FORMAT ('- LP SEL. = ',I2)
      C*
      C**  DISPLAY VARIABLE I/P AREA
      C*
400        CONTINUE
           GO TO (620,405,402), TYPE
402        CONTINUE
                        |
           ----------------------------------------------
           | LPS   =50                                  |
           ----------------------------------------------
                        |
           CALL PTEXT(IGDS,BLINE,VDL,LPS,NULL,3,0,NY-(2*LPSEL)-1)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                 |
              -------
              |411  |
              -------


405          CCNTINUE
                 |
              . * . * .
        . *     IF      * .  T       ------------------------------------
       *    TYPE.NE.        *+++++  |      GO TO 620                    |
       *2                   *       |                                   |
          * .           . *         ------------------------------------
              . * . * .
                F|
                 |
             CALL EXEC(MSG(6))
                 |
             ---------------------------------------------
             | LPS   =50                                 |
             | N     =18                                 |
             | K     =19-LPSEL                           |
             ---------------------------------------------
                 |
              . * . * .
        . *     IF      * .  T       ------------------------------------
       *    LPSEL.EQ        *+++++  |      GO TO 410                    |
       *.1.OR.LPSEL.LT.M    *       |                                   |
          * .           . *         ------------------------------------
              . * . * .
                F|
                 |
             ---------------------------------------------
             | N     =43                                 |
             | K     =M+18-LPSEL                         |
             ---------------------------------------------
                 |
410          CCNTINUE
             CALL PTEXT(IGDS,VDATA,VDL,LPS,NULL,3,N,K)
411          CCNTINUE
             CALL INCL(IGDS,LPS)
             CALL ICURS(IGDS,LPS,NULL,1)
             CALL EXEC(IGDS)
             CALL ENATN(IALEV,32,34)
             CALL RQATN(IALEV,MCODE,2,LPCA,32,34)
             CALL DSATN(IALEV,32,34)
             CALL OMIT(IGDS,LPS)
                 |
              . * . * .
        . *     IF      * .  T       ------------------------------------
       *    MCODE.EQ        *+++++  |      GO TO 500                    |
       *.32                 *       |                                   |
          * .           . *         ------------------------------------
              . * . * .
                F|
                 |
              . * . * .
        . *     IF      * .
       *     (LPCA(4))      *
       *                    *
   |      * .           . *      |
   -          . * . * .          +
   |            0|               |
 -------      -------         -------
 |411  |      |411  |         |420  |
 -------      -------         -------

420          CCNTINUE
```

```
                         IF                T    -----------------------------
                  LPCA(4).                *++++ | GO TO 1050                |
                *EQ.100                   *     -----------------------------
                         F
                         |
                         IF                T    -----------------------------
                  LPCA(4).                *++++ | GO TO 411                 |
                *GT.NOM                   *     -----------------------------
                         F
                         |
              ----------------------------------------
              | LPSEL=LPCA(4)                        |
              ----------------------------------------
                         |
                      | 320 |

          C*

          C**  GET VARIABLE I/P DATA

          C*

  500          CCNTINUE
                         |
                  +-------------+
     +++++++++++  + DO LOOP TO  +
     +            + STMT #  505 +
     +            + N=1,5       +
     +            +-------------+
     +                   |
     +        ----------------------------------------
     +        | RBUF(N) =BLANKS(1)                   |
     +        ----------------------------------------
     +                   |
  505 ++++++++CCNTINUE
                         |
              ----------------------------------------
              | NC    =VDL                           |
              ----------------------------------------
                         |
              ----------------------------------------
              | NS    =-1*NC                         |
              ----------------------------------------
                         |
              CALL GSPRD(IGDS,RBUF,NS,1,NTCODE,LPS,NULL)
                         |
              +------------------------------------+
             +  WRITE (6,510)                     +
            +    RBUF                            +
           +------------------------------------+
                         |
                         |               FORMAT  510 FORMAT (' VAR. ENTRY =
                         |                            ',10A8)
                         IF
                      (NTCODE)
       -                 0                 +
    | 512 |           | 513 |           | 514 |
```

```
                      |
512          | NTCODE =-1*NTCODE                     |
                      |
514          | NC     =NTCODE                        |
                      |
       C*

       C**   CONVERT DATA TO REAL OR INTEGER AS APPLICABLE

       C*

513          CONTINUE
                      |
                  IF          T     GO TO 700
               TYPE.EQ.     *++++
             *3
                 F    |

             CALL CVRTIN(RBUF,NC,X,IERR,TYDATA)
                      |
                  IF          T     GO TO 1030
               IERR.NE.     *++++
             *0.OR.TDATA.NE.TYDAT*
             *A
                 F    |

                  IF          T     GO TO 610
               TDATA.EQ     *++++
             *.2
                 F    |

600          CONTINUE
                      |
                  IF          T     GO TO 1030
               X.LT.LVA     *++++
             *L(LPSEL).OR.X.GT.HV*
             *AL(LPSEL)
                 F    |
                      |
                  |630 |

610          CONTINUE
                      |
             | Z1    =LVAL(LPSEL)                    |
             | Z2    =HVAL(LPSEL)                    |
                      |
                  IF          T     GO TO 1030
               Y.LT.Z3.     *++++
             *OR.Y.GT.Z4
                 F    |
```

```
                     |
                 ---------
                 |630    |
                 ---------


 620          CCNTINUE
                     |
              ------------------------------------------
              | Y      =1                              |
              ------------------------------------------
                     |
              CALL OMIT(IGDS,LPSEL)
 630          CCNTINUE
                     |
              ------------------------------------------
              | OPAR(LPSEL) =X                         |
              ------------------------------------------
                     |
                 ---------
                 |300    |
                 ---------


         C*
         C**   UPDATE LINE OF DATA
         C*
 700          CCNTINUE
              CALL CSMRG(IPAR(I*(LPSEL-1)+1),RBUF,NC)
              CALL PTEXT(IGDS,IPAR(I*(LPSEL-1)+1),MLN,LPSEL,NULL,3,0,NY-2*LPSEL)
                     |
                 ---------
                 |300    |
                 ---------
         C*
         C**  BUILD TEXT DATA SET
         C*
 2000         CCNTINUE
                     |
              ------------------------------------------
              | ECODE=8                                |
              ------------------------------------------
                     |
                     *
                 * *   * *                 ---------------------------
             . *     IF      * .   T       |                         |
            *     MLN .LE.        *++++    |      GO TO 1040         |
            * O .OR. MLN .GT. 80*          |                         |
            *                     *        |                         |
                 * .         . *           ---------------------------
                     * . *
                   F |
              ------------------------------------------
              | ECODE=16                               |
              ------------------------------------------
                     |
                     *
                 * *   * *                 ---------------------------
             . *     IF      * .   T       |                         |
            *     NCM.LE.0         *++++   |      GO TO 1040         |
            *.OR.NCM.GT.49         *       |                         |
                 * .         . *           ---------------------------
                     * . *
                   F |
```

```
                   |
     ---------------------------------------------
     | ECODE=0                                   |
     | IRECAL =0                                 |
     ---------------------------------------------
                   |
     ---------------------------------------------
     | IFLAG=1                                   |
     | X      =MLN                               |
     | X      =(X/4.0)+0.75                      |
     | I      =X                                 |
     ---------------------------------------------
                   |
     CALL SHIFT1
     CALL DICTRY(OPAR(1))
                   |
               +-----------+
               + DO LOOP TO +
  +++++++++++++ STMT #  2003 +
  +            + N = 1, NOM  +
  +            +-----------+
  +                |
  +  ---------------------------------------------
  +  | X      =3920.0-N*80                       |
  +  ---------------------------------------------
  +                |
  +  CALL PTEXT(IGDSC,IPAR(I*(N-1)+1),MLN,NULL,NULL,1,0.0,X)
  +
2003 ++++++++CONTINUE
                                      FORMAT 9999 FORMAT(' DVAR TERMINATE
                                         RECALL: TYPE =',I4,' TDATA =',I4,
               ' MLN =',I4,' NOM =',I4,' NCODE =',I4)
                   |
               --------
               |1052 |
               --------

     C*
     C** SET UP TO UPDATE DATA SET
     C*
3000      CONTINUE
                   |
             . * * .             ---------------------------
         . *    IF     * .  T    |                         |
        *    NCODE.NE     *++++  |   GO TO 1050            |
        *.0               *      |                         |
         * .            . *      ---------------------------
             . * * .
               F |
     ---------------------------------------------
     | ECODE=8                                   |
     ---------------------------------------------
                   |
             . * * .             ---------------------------
         . *    IF     * .  T    |                         |
        *    MLN.LE.0     *++++  |   GO TO 1040            |
        *.OR.MLN.GT.80    *      |                         |
         * .            . *      ---------------------------
             . * * .
               F |
     ---------------------------------------------
     | ECODE=16                                  |
     ---------------------------------------------
                   |
```

```
                   IF                        T   ------------------------------
              NOM.LE.0                    ***** |        GO TO 1040          |
           *.CR.NOM.GT.22                        ------------------------------
                   F

                   IF                        T   ------------------------------
              VDL.LE.0                    ***** |       GO TO 1040           |
           *.OR.VDL.GT.72                        ------------------------------
                   F

          | ECODE=0                                  |
          | X     =MLN                               |
          | X     =(X/4.0)+0.75                      |
          | I     =X                                 |

              +-----------------+
              + DO LOOP TO      +
   ++++++++++++ STMT #   3010   +
   +          + N=1,NOM,1       +
   +          +-----------------+
   +
   +      CALL PTEXT(IGDS,IPAR(I*(N-1)+1),MLN,N,NULL,1,0,NY-2*N)
   +
3010 ++++++++CCNTINUE

               |299  |

       C*
       C**  CONVERSION ROUTINE ERROR RETURN
       C*
1030        CCNTINUE
            CALL EXEC(MSG(5))
            CALL INCL(IGDS,LPS)
          +--------------------------------------+
        +  WRITE (6,1031)                       +
      +--------------------------------------+

               |411  |

                                       FORMAT 1031 FORMAT (' ***  INVALID
                                                    2250 ENTRY   *** ')
       C*
       C**  RETURN TO CALLER
       C*
1040        CCNTINUE
```

```
                 +-----------------------------------------------+
               +  WRITE(6,1041)                                +
             +    NCODE                                      +
           +                                               +
         +-----------------------------------------------+
                           |
                                                            FORMAT 1041 FORMAT(' INVALID I/P
                                                                              PARM(S)',I4)
1050          CCNTINUE
                           |
              --------------------------------------
              | IFLAG=1                            |
              | IRECAL =0                          |
              --------------------------------------
                           |
              CALL TMGDS(IGDS)

              CALL EXEC(MSG(7))
                           |
                  +-------------+
                +   DO LOOP TO  +
      +++++++++++ STMT #   1051 +
      +         +   N=1,7       +
      +           +-------------+
      +                    |
      +
      +       CALL TMGDS(MSG(N))
      +
1051  ++++++++CCNTINUE

1052          CCNTINUE
                           |
              --------------------------------------
              | NCODE=ECODE                        |
              --------------------------------------
                           |
                     * . * . *              -------------------------------------
                 * .     IF      * .  T     |                                   |
              *       TYPE.EQ.        *++++++|   NCODE = NCODE + LPSEL          |
              *3                      *     |                                   |
                 * .             * .        -------------------------------------
                     * . * . *
                       F   |
                           |++++++++++++++++++++++++++++++++++++
                                                            FORMAT 9997 FORMAT(' DVAR TEPMINATE:
              RETURN                                                            NCODE =',I4)

END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
BLOCK DATA

          CCMMCN/GSPD/
             WORK      ,GSP1     ,UNITN    ,I2250    ,IATN     ;
             CAREA     ,NULLV    ,IGDSC    ,IPTNOW   ,MAXPT    ;
             IPT       ,ISYM     ,RWORK1   ,IGDS4    ,IGDS5    ;
             IGDS6     ,IDSP     ,RU       ,HSPB     ,HSPL     ;
             VSPB      ,VSPL     ,NOUT     ,NXD      ,NYD      ;
             XSIZE     ,YSIZE    ,XLL      ,YLL      ,XUR      ;
             YUR       ,XTIC     ,YTIC     ,XDVAL    ,YDVAL    ;
             IKEY1     ,IKEY2    ,IKEY3    ,IKEY4    ,AX       ;
             BX        ,AY       ,BY       ,YLPOS    ,
             LOGXSW    ,LOGYSW   ,NPTF     ,NRGF     ,ERRHD    ;
               FA    ,FB    ,FC    ,GMODE1,GSPIN ,DUM  ,IRECAL

          LCGICAL*1 DUM(2)

          REAL*8 WORK(36)/10*'*********','CONTINUE','---------','---@TABL',
             'E GF CON','TENTS--@','RETURN--','---@NEXT',' P&GE--@',
                18*'        '/

          REAL*4 RU/4C96./    ,HSPB/56./    ,HSPL/84./    ,VSPB/80./   ;
                 VSPL/120./   ,XLL/896./    ,XUR/3984./   ,YUR/3616./  ;
                 YLPOS/3796./,RWORK1/1810./

          INTEGER*4 UNITN/10/     ,OAREA/768/   ,NULLV(1)/-5/,IDSP/0/    ;
                    NOUT/48/      ,NXD/6/       ,NYD/6/       ,GSP1      ;
                    IPT/36/       ,MAXPT/640/   ,IPTNOW/0/

          REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)

          LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD

          LOGICAL*1 FA,FB,FC,GMODE1/.TRUE./,GSPIN/.FALSE./

END
```

```
SUBROUTINE EVPDAY(DPET, EMAET)

      C  DETERMINES DATED PAN EVAPORATION TOTALS

            DIMENSION  DPET(366)
            INTEGER DAY
                         +-------------+
                       +  DO LOOP TO   +
        +++++++++++++  STMT #    100   +
        +              + DAY = 1,5     +
        +                +-------------+
        +                     |
    100 ++++++++| DPET(DAY) =0.00060*EMAET             |
                | DPET(6) =0.00059*EMAET               |
                | DPET(7) =DPET(6)                     |
                ----------------------------------------
                         +-------------+
                       +  DO LOOP TO   +
        +++++++++++++  STMT #    101   +
        +              + DAY = 8,10    +
        +                +-------------+
    101 ++++++++| DPET(DAY) =0.00058*EMAET             |
                ----------------------------------------
                         +-------------+
                       +  DO LOOP TO   +
        +++++++++++++  STMT #    102   +
        +              + DAY = 11,16   +
        +                +-------------+
    102 ++++++++| DPET(DAY) =0.00057*EMAET             |
                | DPET(17) =DPET(8)                    |
                ----------------------------------------
                         +-------------+
                       +  DO LOOP TO   +
        +++++++++++++  STMT #    103   +
        +              + DAY = 18,20   +
        +                +-------------+
    103 ++++++++| DPET(DAY) =DPET(6)                   |
                ----------------------------------------
                         +-------------+
                       +  DO LOOP TO   +
        +++++++++++++  STMT #    104   +
        +              + DAY = 21,32   +
        +                +-------------+
    104 ++++++++| DPET(DAY) =DPET(1)                   |
                | DPET(33) =0.00061*EMAET              |
                ----------------------------------------
                         +-------------+
                       +  DO LOOP TO   +
        +++++++++++++  STMT #    105   +
        +              + DAY = 34,38   +
        +                +-------------+
    105 ++++++++| DPET(DAY) =0.00062*EMAET             |
                | DPET(39) =0.00063*EMAET              |
                | DPET(40) =DPET(39)                   |
                | DPET(41) =0.00064*EMAET              |
                | DPET(42) =0.00065*EMAET              |
                | DPET(43) =0.00066*EMAET              |
                ----------------------------------------
```

```
                    +--------------+
                    + DO LOOP TO   +
         ++++++++++++ STMT #   106 +
         +          + DAY = 44,50  +
         +          +--------------+
         +
106 +++++++++| DPET(DAY) =0.00067*EMAET                      |

                    +--------------+
                    + DO LOOP TO   +
         ++++++++++++ STMT #   107 +
         +          + DAY = 51,55  +
         +          +--------------+
         +
107 +++++++++| DPET(DAY) =0.00068*EMAET                      |
             | DPET(56) =0.00069*EMAET                       |

                    +--------------+
                    + DO LOOP TO   +
         ++++++++++++ STMT #   108 +
         +          + DAY = 57,61  +
         +          +--------------+
         +
108 +++++++++| DPET(DAY) =0.00070*EMAET                      |
             | DPET(62) =0.00071*EMAET                       |
             | DPET(63) =0.00072*EMAET                       |
             | DPET(64) =DPET(63)                            |
             | DPET(65) =0.00073*EMAET                       |
             | DPET(66) =0.00074*EMAET                       |
             | DPET(67) =0.00075*EMAET                       |
             | DPET(68) =0.00076*EMAET                       |
             | DPET(69) =0.00077*EMAET                       |
             | DPET(70) =DPET(69)                            |
             | DPET(71) =0.00078*EMAET                       |
             | DPET(72) =DPET(71)                            |
             | DPET(73) =0.00079*EMAET                       |
             | DPET(74) =DPET(73)                            |
             | DPET(75) =0.00080*EMAET                       |
             | DPET(76) =0.00081*EMAET                       |
             | DPET(77) =0.00082*EMAET                       |
             | DPET(78) =0.00084*EMAET                       |
             | DPET(79) =0.00086*EMAET                       |
             | DPET(80) =0.00088*EMAET                       |
             | DPET(81) =0.00090*EMAET                       |
             | DPET(82) =0.00092*EMAET                       |
             | DPET(83) =0.00094*EMAET                       |

             | DPET(84) =0.00097*EMAET                       |
             | DPET(85) =0.00099*EMAET                       |
             | DPET(86) =0.00102*EMAET                       |
             | DPET(87) =0.00106*EMAET                       |
             | DPET(88) =0.00109*EMAET                       |
             | DPET(89) =0.00113*EMAET                       |
             | DPET(90) =0.00118*EMAET                       |
             | DPET(91) =0.00122*EMAET                       |
             | DPET(92) =0.00128*EMAET                       |
             | DPET(93) =0.00132*EMAET                       |
             | DPET(94) =0.00137*EMAET                       |
             | DPET(95) =0.00142*EMAET                       |
             | DPET(96) =0.00147*EMAET                       |
             | DPET(97) =0.00151*EMAET                       |
             | DPET(98) =0.00157*EMAET                       |
             | DPET(99) =0.00163*EMAET                       |
             | DPET(100) =0.00168*EMAET                      |
             | DPET(101) =0.00173*EMAET                      |
             | DPET(102) =0.00178*EMAET                      |
             | DPET(103) =0.00185*EMAET                      |
             | DPET(104) =0.00193*EMAET                      |
             | DPET(105) =0.00201*EMAET                      |
             | DPET(106) =0.00208*EMAET                      |
             | DPET(107) =0.00214*EMAET                      |
             | DPET(108) =0.00221*EMAET                      |
             | DPET(109) =0.00227*EMAET                      |
             | DPET(110) =0.00234*EMAET                      |
             | DPET(111) =0.00241*EMAET                      |
             | DPET(112) =0.00249*EMAET                      |
```

```
DPET(113) =0.00256*EMAET
DPET(114) =0.00262*EMAET
DPET(115) =0.00268*EMAET
DPET(116) =0.00276*EMAET
DPET(117) =0.00281*EMAET
DPET(118) =0.00287*EMAET
DPET(119) =0.00293*EMAET
DPET(120) =0.00299*EMAET
DPET(121) =0.00305*EMAET
DPET(122) =0.00310*EMAET
DPET(123) =0.00317*EMAET
DPET(124) =0.00322*EMAET
DPET(125) =0.00328*EMAET
DPET(126) =0.00333*EMAET
DPET(127) =0.00338*EMAET
DPET(128) =0.00344*EMAET
DPET(129) =0.00348*EMAET
DPET(130) =0.00354*EMAET
DPET(131) =0.00359*EMAET
DPET(132) =0.00365*EMAET
DPET(133) =0.00370*EMAET
DPET(134) =0.00374*EMAET
DPET(135) =0.00378*EMAET
DPET(136) =0.00382*EMAET
DPET(137) =0.00387*EMAET
DPET(138) =0.00391*EMAET
DPET(139) =0.00394*EMAET
DPET(140) =0.00399*EMAET
DPET(141) =0.00402*EMAET
DPET(142) =0.00407*EMAET
DPET(143) =0.00411*EMAET
DPET(144) =0.00417*EMAET
DPET(145) =0.00420*EMAET
DPET(146) =0.00426*EMAET
DPET(147) =0.00430*EMAET
DPET(148) =0.00436*EMAET
DPET(149) =0.00440*EMAET
DPET(150) =0.00446*EMAET
DPET(151) =0.00450*EMAET
DPET(152) =0.00455*EMAET
DPET(153) =0.00460*EMAET
DPET(154) =0.00466*EMAET
DPET(155) =0.00470*EMAET
```

```
DPET(156) =0.00473*EMAET
DPET(157) =0.00478*EMAET
DPET(158) =0.00482*EMAET
DPET(159) =0.00487*EMAET
DPET(160) =0.00491*EMAET
DPET(161) =0.00495*EMAET
DPET(162) =0.00500*EMAET
DPET(163) =0.00504*EMAET
DPET(164) =0.00508*EMAET
DPET(165) =0.00510*EMAET
DPET(166) =0.00512*EMAET
DPET(167) =0.00514*EMAET
DPET(168) =0.00515*EMAET
DPET(169) =0.00517*EMAET
DPET(170) =0.00519*EMAET
DPET(171) =0.00520*EMAET
DPET(172) =0.00521*EMAET
DPET(173) =DPET(172)
DPET(174) =DPET(172)
DPET(175) =0.00522*EMAET
DPET(176) =0.00523*EMAET
DPET(177) =0.00524*EMAET
DPET(178) =0.00525*EMAET
DPET(179) =0.00527*EMAET
DPET(180) =0.00528*EMAET
DPET(181) =DPET(180)
DPET(182) =0.00529*EMAET
DPET(183) =0.00530*EMAET
DPET(184) =DPET(183)
DPET(185) =0.00531*EMAET
DPET(186) =0.00532*EMAET
DPET(187) =0.00533*EMAET
DPET(188) =0.00534*EMAET
DPET(189) =DPET(188)
DPET(190) =0.00535*EMAET
DPET(191) =0.005[illegible]5*EMAET
DPET(192) =0.00537*EMAET
DPET(193) =0.00538*EMAET
DPET(194) =DPET(193)
DPET(195) =0.00539*EMAET
DPET(196) =0.00540*EMAET
DPET(197) =DPET(196)
```

```
          | DPET(198) =0.00541*EMAET                         |
          | DPET(199) =0.00542*EMAET                         |
          | DPET(200) =0.00543*EMAET                         |
          | DPET(201) =0.00545*EMAET                         |
          | DPET(202) =0.00546*EMAET                         |
          | DPET(203) =0.00547*EMAET                         |
          | DPET(204) =0.00548*EMAET                         |
          | DPET(205) =0.00549*EMAET                         |
          | DPET(206) =0.00550*EMAET                         |
          | DPET(207) =0.00551*EMAET                         |
          | DPET(208) =0.00552*EMAET                         |
          | DPET(209) =0.00553*EMAET                         |
          | DPET(210) =0.00555*EMAET                         |
          | DPET(211) =0.00557*EMAET                         |
          | DPET(212) =0.00558*EMAET                         |
          | DPET(213) =0.00560*EMAET                         |
          | DPET(214) =DPET(213)                             |
          | DPET(215) =0.00561*EMAET                         |
          | DPET(216) =0.00562*EMAET                         |
          | DPET(217) =0.00563*EMAET                         |
          | DPET(218) =0.00565*EMAET                         |
          | DPET(219) =0.00567*EMAET                         |
          | DPET(220) =DPET(219)                             |
          ----------------------------------------------------
                       |
                +-------------+
                +  DO LOOP TO +
  +++++++++++++++  STMT #  109 +
  +             + DAY = 221,226+
  +             +-------------+
  +                    |
  +       ----------------------------------------------------
109 ++++++| DPET(DAY) =0.00568*EMAET                         |
          ----------------------------------------------------
                       |
                +-------------+
                +  DO LOOP TO +
  +++++++++++++++  STMT #  110 +
  +             + DAY = 227,229+
  +             +-------------+
  +                    |
  +       ----------------------------------------------------
110 ++++++| DPET(DAY) =DPET(219)                             |
          | DPET(230) =0.00566*EMAET                         |
          | DPET(231) =0.00564*EMAET                         |
          | DPET(232) =DPET(217)                             |
          | DPET(233) =DPET(216)                             |
          | DPET(234) =DPET(213)                             |
          | DPET(235) =0.00559*EMAET                         |
          | DPET(236) =DPET(211)                             |
          | DPET(237) =DPET(210)                             |
          | DPET(238) =DPET(209)                             |
          | DPET(239) =DPET(206)                             |
          | DPET(240) =DPET(203)                             |
          | DPET(241) =DPET(199)                             |
          | DPET(242) =DPET(193)                             |
          | DPET(243) =DPET(190)                             |
          | DPET(244) =DPET(185)                             |
          | DPET(245) =DPET(179)                             |
          | DPET(246) =DPET(175)                             |
          | DPET(247) =DPET(169)                             |
          | DPET(248) =0.00511*EMAET                         |
          | DPET(249) =DPET(163)                             |
          | DPET(250) =0.00497*EMAET                         |
          | DPET(251) =0.00490*EMAET                         |
          | DPET(252) =DPET(158)                             |
          | DPET(253) =0.00476*EMAET                         |
          | DPET(254) =0.00468*EMAET                         |
          | DPET(255) =0.00461*EMAET                         |
          | DPET(256) =0.00454*EMAET                         |
          | DPET(257) =DPET(150)                             |
          | DPET(258) =0.00437*EMAET                         |
          | DPET(259) =0.00427*EMAET                         |
          | DPET(260) =0.00418*EMAET                         |
          | DPET(261) =DPET(142)                             |
          | DPET(262) =0.00397*EMAET                         |
          | DPET(263) =DPET(137)                             |
          | DPET(264) =0.00377*EMAET                         |
          | DPET(265) =0.00367*EMAET                         |
          | DPET(266) =0.00356*EMAET                         |
          | DPET(267) =0.00347*EMAET                         |
          | DPET(268) =0.00337*EMAET                         |
          | DPET(269) =0.00329*EMAET                         |
          | DPET(270) =DPET(124)                             |
          | DPET(271) =0.00315*EMAET                         |
          | DPET(272) =0.00308*EMAET                         |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
DPET(273) =0.00303*EMAET
DPET(274) =0.00300*EMAET
DPET(275) =0.00298*EMAET
DPET(276) =0.00294*EMAET
DPET(277) =0.00290*EMAET
DPET(278) =0.00286*EMAET
DPET(279) =0.00283*EMAET
DPET(280) =0.00279*EMAET
DPET(281) =DPET(116)
DPET(282) =0.00271*EMAET
DPET(283) =DPET(115)
DPET(284) =DPET(114)
DPET(285) =0.00259*EMAET
DPET(286) =0.00254*EMAET
DPET(287) =0.00252*EMAET
DPET(288) =0.00247*EMAET
DPET(289) =0.00244*EMAET
DPET(290) =0.00239*EMAET
DPET(291) =DPET(110)
DPET(292) =0.00230*EMAET
DPET(293) =0.00225*EMAET
```

```
DPET(294) =0.00222*EMAET
DPET(295) =0.00217*EMAET
DPET(296) =0.00213*EMAET
DPET(297) =0.00210*EMAET
DPET(298) =0.00206*EMAET
DPET(299) =0.00200*EMAET
DPET(300) =0.00197*EMAET
DPET(301) =0.00194*EMAET
DPET(302) =0.00189*EMAET
DPET(303) =0.00186*EMAET
DPET(304) =0.00183*EMAET
DPET(305) =0.00180*EMAET
DPET(306) =0.00177*EMAET
DPET(307) =0.00174*EMAET
DPET(308) =0.00172*EMAET
DPET(309) =DPET(100)
DPET(310) =DPET(99)
DPET(311) =0.00160*EMAET
DPET(312) =0.00156*EMAET
DPET(313) =0.00152*EMAET
DPET(314) =0.00149*EMAET
DPET(315) =0.00146*EMAET
DPET(316) =DPET(95)
DPET(317) =0.00138*EMAET
DPET(318) =0.00135*EMAET
DPET(319) =0.00131*EMAET
DPET(320) =0.00127*EMAET
DPET(321) =0.00124*EMAET
DPET(322) =0.00120*EMAET
DPET(323) =DPET(90)
DPET(324) =0.00116*EMAET
DPET(325) =DPET(89)
DPET(326) =0.00110*EMAET
DPET(327) =0.00107*EMAET
DPET(328) =0.00104*EMAET
DPET(329) =DPET(86)
DPET(330) =0.00100*EMAET
DPET(331) =0.00098*EMAET
DPET(332) =0.00097*EMAET
DPET(333) =0.00095*EMAET
DPET(334) =0.00093*EMAET
DPET(335) =DPET(81)
DPET(336) =DPET(80)
DPET(337) =0.00087*EMAET
DPET(338) =DPET(79)
DPET(339) =DPET(78)
DPET(340) =DPET(77)
DPET(341) =DPET(75)
DPET(342) =DPET(73)
DPET(343) =DPET(71)
DPET(344) =DPET(71)
DPET(345) =DPET(69)
DPET(346) =DPET(68)
DPET(347) =DPET(66)
DPET(348) =DPET(63)
DPET(349) =DPET(62)
DPET(350) =DPET(57)
DPET(351) =DPET(57)
DPET(352) =DPET(56)
```

```
                  +------------+
                  + DO LOOP TO  +
      +++++++++++ STMT #    111 +
      +           + DAY = 353,355+
      +           +------------+
      +                 |
      +                 |
111   ++++++++| DPET(DAY) =DPET(51)                     |
              | DPET(356) =DPET(44)                     |
              ------------------------------------------
                        |
              ------------------------------------------
              | DPET(357) =DPET(44)                     |
              | DPET(358) =DPET(42)                     |
              | DPET(359) =DPET(41)                     |
              | DPET(360) =DPET(39)                     |
              | DPET(361) =DPET(34)                     |
              | DPET(362) =DPET(33)                     |
              ------------------------------------------
                        |
                  +------------+
                  + DO LOOP TO  +
      +++++++++++ STMT #    112 +
      +           + DAY = 363,365+
      +           +------------+
      +                 |
      +                 |
112   ++++++++| DPET(DAY) =DPET(1)                      |
              | DPET(366) =DPET(57)                     |
              ------------------------------------------
                        |
              RETURN

END
```

```
PROGRAM FINDS THE INTEGRAL OF A GIVEN HYDROGRAPH AND PLOTS THE
C      INPUT DATA.
C
C      DATA INPUT IN HRS.FT.
C      DATA PLOTTED IN HRS.FT. OR HRS.SEC.FT.
C      DATA INTEGRATED IN DAY.FT. OR DAY.SEC.FT.
C      IFLAG= 1  TRAPEZOIDAL INTEGRATION SCHEME USED.
C      IFLAG= 2  SIMPSON'S 1/3 INTEGRATION SCHEME USED.
C      IPLOT= 0  1403 PRINTER PLOTS ONLY.
C      IPLOT= 1   BOTH 1403 PRINTER PLOTS AND SC4020 PLOTS.
C      LOOKUP= -1  NO TABLE LOOKUP REQUIRED FOR CURRENT CASE.
C      LOOKUP=  0  PREVIOUS TABLE REQUIRED FOR CURRENT CASE.
C      LOOKUP=  1  NEW TABLE REQUIRED FOR CURRENT CASE.
C
      SUBROUTINE INT(NQ,Q,DT,AREAI,LOOKUP,IPLOT,NFP,MSBDIC,MPDAY,DIFFS,
      DIFFMP,DIFFP,DIFFPP,SUM,DIFFRA,DIFFRP,APREC,DIFFPR,DIFPP,QOUT,
      QMAX,PHRO)

      COMMON/PLOTC/DRSF,DSSF,CONOPT,THSFD,TMSTF,STMROS(121,6),DPY,TITLE,
      KFLAG,IDFLAG,
      IENDFG,STUDY(2),PEAKS,PHRS,NSPTS,THSFDM,TFMAXD,TMRTF,JPLOT,
        NCTRI,CTRI,FIRR,RICY,DPSE,BODFSM,SPBFLW,SPTWCC,SP1,ELDIF,
        XDNFS,FFOR,FFSI,MRNSM,DSMGH,PXCSA,RMPF,RGPMB,AREA,FIMP,
       SATRI,UHFA,
      MNRD,
        FWTR,VINTMR,BUZC,SUZC,LZC,ETLF,SUBWF,GWETF,SIAC,BMIR,
        BIVF,OFSS,OFSL,OFMN,OFMNIS,IFRC,CSRX,FSRX,CHCAP,EXOPV,
        BFNLR,BFRC,GWS,UZS,LZS,BFNX,IFS,BFHRC,BFRL,BFNRL,BFNHR,IFPRC,
        IFRL,LSHFT,NBTRI,FNTRI,MXTRI,NCSTRI,BIRI,TFCFS,EPACT,EPER,
        TPLR,VINTCR,HSF,NRTRI,SPIF,CBF,SPDR,CFUS,OFUSIS,OFR,OFRIS,PEIS,
        RHFO,URHF,AMIF,AMNET,AMPET,AMSNE,AMFSIL,SASFX,SARAX,SRX,VWIN,
        WCFS,RHFMC,SSRT,OFRF,OFRFIS,EQDF,EQDFIS,SOFRF,SOFRFI,
        SDEPTH,MULTI,ID,ASM,WT4AM,WT4PM,SAX,TANSM,SPTW,STMD,SFMD,ASMRG,
      DEPEND(2),VARIN(2),NPIS,JULDI,IYR,TODARY(5,1),
      TOMARY(7,1),TOSARY(5,6,1),TSDARY(6,1),TSMARY(8,1),
      TSSARY(3,6,1),TSMCRY(1),TSDCRY(1),TSRARY(1,6,1),
      TORARY(1,6,1),
      DRSFT(366),DSSFT(366),MI,NI,MULT,TMRTFT(12),TMSTFT(12)

      COMMON/COMMA/EMBFNX,EMGWS,EMIFS,EMLZS,EMSIAM,EMUZC,EMUZS,TMBF,
      TMIF,TMPREC,TMSE,CRFMI,DDIW,DMNT,DMXT,DRGPM,DRHP,DRSGP,DPET,EDLZS,
      EPCM,SERA,SERR,SESF,SOER,THSF,TMFSIL,TMNET,TMOF,TMPET,TMRPM,TMSNE,
      TMSTFI,T200FH,T20PRH,TMRTFI,JULDAT,
      TFMAXY,UZC,AETX,DAY,NSGRD,AEX90,SIAM,NDSDP,RGPM,NDSDR,YR1,
      TRHF,
      SINDEX,INDEX,AEX96,MAXI,YR2,BYLZS,BYIFS,BYUZS

      DIMENSION TITLE(18),Q(337),T(337),TABT(18)
                ,FLDT(18),FLDQ(18),IDF(18),TEMP(1540)
                ,YLAB(13)     ,FLD(6),A(10),SQ(500),ANSQ(3),ANSS(3),
        NSPTS(6),SUBT(18),DRSF(366),DSSF(366),TITLA(18),CONOPT(16),
      THSFD(744,3),PEAKS(6),THSFDM(6),PHRS(6),TMRTF(12),TMSTF(15,3),
        CTRI(99),FIRR(15),RICY(37),DPSE(366),BTRI(99),
       CRFMI(22),DDIW(366),DMNT(366),DMXT(366),DRGPM(366),DRHP(366,24),
       EMBFNX(15,3),EMGWS(15,3),EMIFS(15,3),EMLZS(15,3),EMSIAM(15,3),
      DRSGP(366),DPET(366),EDLZS(366),SERA(22),SERR(22),SESF(22),
      SCER(22),
       EMUZC(15,3),EMUZS(15,3),EPCM(12),
                 THSF(24),TMBF(15,3),TMFSIL(12),TMIF(15,3),TMNET(12),
       TMOF(15,3),TMPET(12),TMPREC(15,3),TMRPM(12),TMSE(15,3),
                 TMSNE(12),TMSTFI(15,3),T200FH(21),T20PRH(21),
      TMRTFI(12),JULDAT(6),TFMAXY(366),
       SATRI(99),UHFA(99),APREC(3),QOUT(15)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      INTEGER  DIFFP,STPTS,PHRS,FLDY(18),STUDY,TITLA,TABT,END
      INTEGER TOMARY,TSMARY,TODARY,TSDARY,TOSARY,TSSARY
      INTEGER FLDP(18),CONOPT,PCHARP(2),PCHARF(3),PCHAR(3),SINDEX,
     MPDAY(15),FLDQ,FLDT
      INTEGER    ANSC,ANSS,PHRO,SUBT
      REAL SUM(3)
      DATA PCHARP/38,50/
      DATA PCHARF/54,57,22/
      INTEGER PFCHAR(3)/-230,-233,198/,PPCHAR(2)/-214,226/
      INTEGER PRCHAR(3)
      DATA SUBT/'REF ','R/O ',3*4H    ,' IN ','<TIM',
     'E(HR',')> S','IM R','/O  ',3*4H    ,' IN ',3*4H    /
      DATA END/'END '/
      DATA BLK /4H    /
      DATA FLDQ/'W/C ','VS N','/P V','S F/','C HO','URLY',' R/O',
     11*4H    /
      DATA FLDP/'HOUR','LY  ','PREC','IPAT','ATIO','N   ','(IN)',
     11*4H    /
      DATA FLDY/'HOUR','LY O','BS A','ND S','IM R','/O (','CFS)',
     11*4H    /
      DATA FLDT/6*4H    ,'TIME','(HRS',')   ',9*4H    /
      DATA FLD /'E(FT',')   ','    ','E(SE','C.FT','.)  '/
C
                          |
              --------------------------------
  9           | IERR =0                      |
              | INDEXM =0                    |
              | IDIAG=1                      |
              --------------------------------
                          |
                 +------------+
                 + DO LOOP TO +
      +++++++++++ STMT #  180 +
      +          + I=1,1540   +
      +          +------------+
      +                   |
      +                   |
              --------------------------------
180   ++++++++| TEMP(I) =0.0
              --------------------------------
                          |
            +-----------------------------------------+
           +  READ(5,500)                            +
          +   (IDF(I),I=1,18)                       +
         +                                         +
        +-----------------------------------------+
                          |
                          |                     FORMAT  500 FORMAT(18A4)
                 +------------+
                 + DO LOOP TO +
      +++++++++++ STMT #   16 +
      +          + I=1,18     +
      +          +------------+
      +                   |
      +               . * * .              ----------------------
      +          . *    IF     * .   T     |   GO TO 10         |
      +        *    IDF(I).E        *+++++ |                    |
      +        *Q.BLK               *      |                    |
      +          * .           . *         ----------------------
      +               *  |  *
      +                F |
 16   ++++++++CONTINUE
```

```
                    +--------------+
                    +  DO LOOP TO  +
         +++++++++++  STMT #  4024 +
         +          + I=1,2        +
         +          +--------------+
         +                 I
         +   C      IDF(14+I)=STUDY(I)
         +
4024     ++++++++CONTINUE
                 CALL PLOTID(IDF,1,9)
                       I
                    -------
                    I20   I
                    -------


10               CONTINUE
             C
                       I
20            +------------------------------------------------+
             +  READ(5,500,END=777)                           +
            +   (TITLA(I),I=1,18)                            +
           +                                                +
          +------------------------------------------------+
                       I
                       I
                       *
                   *       *              T  ------------------------------
                *     IF      *              I   GO TO 777                I
              *    TITLA(1)     *   +++++    I                            I
              *.EQ.END          *            I                            I
                *             *              ------------------------------
                   *       *
                       *
                       F
                       I
                    +--------------+
                    +  DO LOOP TO  +
         +++++++++++  STMT #  4023 +
         +          + I=1,2        +
         +          +--------------+
         +                 I
         +          ---------------------------------------
         +          I TITLA(16+I) =STUDY(I)               I
         +          ---------------------------------------
4023     ++++++++CONTINUE
                           I
                    +--------------+
                    +  DO LOOP TO  +
         +++++++++++  STMT #    59 +
         +          + I=1,NFP      +
         +          +--------------+
         +                 I
         +          ---------------------------------------
         +          I APREC(I) =0.0                       I
         +          ---------------------------------------
         +                 I
59       ++++++++CONTINUE
                           I
                    +--------------+
                    +  DO LOOP TO  +
         +++++++++++  STMT #    61 +
         +          + K=1,NFP      +
         +          +--------------+
         +                 I
         +          +--------------+
         +          +  DO LOOP TO  +
         +++++++++++  STMT #    63 +
         +          + I=1,MAXI     +
         +          +--------------+
         +                 I
         +          ---------------------------------------
         +          I APREC(K) =TMPREC(I,K)+APREC(K)      I
         +          ---------------------------------------
```

```
   +
   +
63 ++++++++CCNTINUF
   +
61 ++++++++CCNTINUE
                    |
              +------------+
              + DO LOOP TO +
   +++++++++++ STMT #   65 +
   +          + I=1,NFP    +
   +          +------------+
   +                |
   +          +------------+
   +          + DO LOOP TO +
   +++++++++++ STMT #   67 +
   +          + J=1,MAXI   +
   +          +------------+
   +                |
   +     ----------------------------------------
   +     | TMSTFI(J,I) =(TMSTF(J,I))/VWIN       |
   +     | TMOF(J,I) =TMSTFI(J,I)-TMIF(J,I)     |
   +     |          -TMBF(J,I)+TMSE(J,I)        |
   +     ----------------------------------------
   +                |
67 ++++++++CCNTINUE
   +
65 ++++++++CCNTINUE
                    |
              +------------+
              + DO LOOP TO +
   +++++++++++ STMT #  322 +
   +          + I=1,NFP    +
   +          +------------+
   +                |
   +     ----------------------------------------
   +     | THSFDT =THSFD(1,I)                   |
   +     ----------------------------------------
   +                |
   +     ----------------------------------------
   +     | PEAKS(I) =THSFD(1,I)                 |
   +     | PHRS(I) =1                           |
   +     ----------------------------------------
   +                |
   +  C
   +                |
   +          +------------+
   +          + DO LOOP TO +
   +++++++++++ STMT #  320 +
   +        --+ J=2,SINDEX +
   +          +------------+
   +                |
   +          * * * * *             -------------------------
   +       *      IF      *    T    |                       |
   +      *    THSFD(J,    *+++++++ |     GO TO 320         |
   +      *I).LE.THSFDT    *        |                       |
   +       *              *         -------------------------
   +          * * * * *
   +            F   |
   +     ----------------------------------------
   +     | PEAKS(I) =THSFD(J,I)                 |
   +     | THSFDT =THSFD(J,I)                   |
   +     | PHRS(I) =J                           |
   +     ----------------------------------------
   +                |
320 ++++++++CCNTINUE
322 ++++++++CCNTINUE
                    |
              +------------+
330           + DO LOOP TO +
   +++++++++++ STMT #  332 +
   +          + I=1,NFP    +
   +          +------------+
   +                |
   +     ----------------------------------------
   +     | THSFDT =THSFD(1,I)                   |
   +     ----------------------------------------
```

```
        DO LOOP TO
        STMT #    334
        K=2,SINDEX

334     THSFDT =AMIN1(THSFDT,THSFD(K,I))
        THSFDM(I) =THSFDT

332     CCNTINUE

        IF
        LOOKUP.L          T
        E.0               ----> GO TO 40
        F

        KS    =0

        READ(5,500)
        (TABT(I),I=1,18)

        DO LOOP TO
        STMT #   8003
        I=1,2

        TABT(16+I) =STUDY(I)

8003    CCNTINUE

        DO LOOP TO
        STMT #     15
        J=1,50

        DO LOOP TO
        STMT #    505
        K=1,10

        CALL READ(A(K))

        IF
        A(K).EQ.          T
        999.0             ----> GO TO 112
        F

        L     =K

505     CCNTINUE
113     CCNTINUE
```

```
      C
      C     STORE 10 VALUES IN TABLE
                    |
                DO LOOP TO
                STMT #    12
                M=1,L
                    |
            | KS    =KS+1                       |
                    |
                   IF                T     NPTS=KS
                A(M).NE.
            0.
                 F  |
            | DEPEND(KS) =A(M)                  |
                    |
12    CCNTINUE
      C
      C     TEST FOR BLANK CARD AT END
                    |
            | TEST =999.0                       |
                    |
      C     IF TEST=0,YOU HAVE THE BLANK CARD AT THE END
                    |
                   IF                T     GO TO 14
                TEST.EQ.
            A(K)
                 F  |
15    CCNTINUE
14          CONTINUE
      C     NPTS IS NUMBER OF POINTS,IS SET AFTER 14

              CALL READ(VARIN(1),XINC)
                    |
                DO LOOP TO
                STMT #    30
                I=2.NPTS
                    |
30          | VARIN(I) =VARIN(I-1)+XINC         |
                    |
40          | T(1) =0.0                         |
                    |
                   IF                T     NQ=SINDEX
                CONOPT(1
            5).NE.2
                 F  |
```

ORIGINAL PAGE IS OF POOR QUALITY

```
                      +--------------+
                    +  DO LOOP TO     +
+++++++++++++++++++++  STMT #     60   +
+                   +  I=2,NQ          +
+                     +--------------+
+                           |
+                           |
+                   -----------------------------------------------
60 ++++++++++++++++| T(I) =T(I-1)+DT                               |
                    -----------------------------------------------
                            |
                                              FORMAT   606 FORMAT('1',18A4)
                            |
                                         ----------------------------------
                          IF          T  |                                |
                 *     CONOPT(1      *++++|   GO TO 75                    |
                 *5).NE.2            *   |                                |
                            F            ----------------------------------
                            |
                                              FORMAT   613 FORMAT(' WATER BASIN
                                                       AREA(SQ. MI.)=',F7.1)
                                         ----------------------------------
                          IF          T  |   GO TO 75                     |
                 *     LOOKUP.L      *++++|                               |
                 *T.O                *   |                                |
                            F            ----------------------------------
                            |

 C      CONVERT FT. OF GAUGE HEIGHT INTO SEC.FT. OF GAUGE HEIGHT.

                      +--------------+
                    +  DO LOOP TO     +
+++++++++++++++++++++  STMT #     70   +
+                   +  I=1,NQ          +
+                     +--------------+
+                           |
+                   -----------------------------------------------
70 ++++++++++++++++| Q(I) =TABLE(VARIN,DEPEND,NPTS,Q(I))            |
                    -----------------------------------------------
                            |
 C
                            |
                    -----------------------------------------------
75                 | IFLAG=2                                       |
                    -----------------------------------------------
                            |
                                         ----------------------------------
                          IF          T  |    IFLAG= 1                    |
                 *     MOD(NQ,2      *++++|                               |
                 *).EQ.0             *   |                                |
                            F            ----------------------------------
                            |+++++++++++++++++++++++++++++++
                    -----------------------------------------------
                   | H     =DT/24.                                 |
                    -----------------------------------------------
                            |
 C
                            |
                                         ----------------------------------
                          IF          T  |   GO TO 100                    |
                 *     LOOKUP.L      *++++|                               |
                 *T.O                *   |                                |
                            F            ----------------------------------
                            |
              +-----------------------------------------------+
            +  WRITE(6,606)                                  +
           +   (TABT(I),I=1,18)                             +
          +                                                +
        +-----------------------------------------------+
                            |
```

```
WRITE(6,612)
NPTS

                    FORMAT  612 FORMAT('ONUMBER OF POINTS
                                IN RATING TABLE=',I6)

WRITE(6,607)
(VARIN(I),I=1,NPTS)

                    FORMAT  607 FORMAT('0INDEPENDENT
                                VARIABLES'/(12E11.4))

WRITE(6,608)
(DEPEND(I),I=1,NPTS)

                    FORMAT  608 FORMAT(//'0DEPENDENT
                                VARIABLES'/(12E11.4))

WRITE(6,609)
(T(I),Q(I),I=1,NQ)

                    FORMAT  609 FORMAT(//'0TIME(HRS.)'
          ,5X,'OBSERVED DISCHARGE(SEC.FT. OF GAUGE HEI
GHT)' /(F9.2,F22.4))

    DO LOOP TO
    STMT #     90
    I=3,5

90  FLDQ(I) =FLD(I+1)

100
    IF
    CONOPT(1
    5).NE.2          T    GO TO 25
    F

    QMAX =Q(1)
    PHRO =1

    DO LOOP TO
    STMT #    161
    I=1,NQ

    IF
    Q(I).LE.
    QMAX             T    GO TO 161
    F
```

```
        +              |
        +        ------------------------------------------
        +       | QMAX =Q(I)                               |
        +       | PHRO =I                                  |
        +        ------------------------------------------
        +              |
161     ++++++++CONTINUE
                       |
                 ------------------------------------------
                | QMIN =QMAX                               |
                 ------------------------------------------
                       |
                  +------------+
                + DO LOOP TO    +
       ++++++++++++ STMT #   160 +
       + ---------+ I=1,NQ  -----+
       +          +------------+
       +               |
       +               |
160    ++++++++| QMIN =AMIN1(QMIN,Q(I))                    |
               | QMAP =AINT(QMAX+1.)                       |
                 ------------------------------------------
                       |
C-
C PRINTER PLOTS
                       |
                 ------------------------------------------
                | NPPTS=NQ                                 |
                | INDEXM =1                                |
                | STPTS=SINDEX                             |
                 ------------------------------------------
                       |
                  +------------+
                + DO LOOP TO    +
       ++++++++++++ STMT #   380 +
       +        + K=1,STPTS     +
       +          +------------+
       +               |
380    ++++++++| SQ(K)=THSFD(K,INDEXM)                     |
               | PEAK =AINT(PEAKS(INDEXM)+1)               |
                 ------------------------------------------
                       |
                  * * *           ----------------------------
             *     IF      *   T |      IFLAG=1              |
            *   MOD(STPT    *++++|                           |
            *S,2).EQ.0      *    |                           |
             *             *      ----------------------------
                  * * *
                    F  |
                       |++++++++++++++++++++++++++++++++++++++
CALL INTEG(IFLAG,H,Q,SSUM,STPTS)
                       |
               | SUM(2) =SSUM/(26.9*AREAI)                 |
                 ------------------------------------------
                       |
CALL INTEG(IFLAG,H,SQ,SSUM,STPTS)
                       |
                 ------------------------------------------
               | SUM(1) =SSUM/(26.9*AREAI)                 |
               | KNT   =1                                  |
               | M     =1                                  |
               | SSUM  =SUM(2)                             |
                 ------------------------------------------
                       |
                  +------------+
                + DO LOOP TO    +
       ++++++++++++ STMT #     7 +
       +        + I=1,2         +
       +          +------------+
       +               |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      +      -----------------------------------------------------
      +      | PCHAR(I) =PCHARP(I)                                 |
      +      | PRCHAR(I) =PPCHAR(I)                                |
      +      -----------------------------------------------------
      +                    |
      +                    |
   7 +++++++++CONTINUE
                           |
                           .
                     . *   *  .          ---------------------------
                 . *     IF     * .   T  |        QMAP=PEAK          |
               *      QMAP.LT.      *++++|                           |
               *PEAK                *    |                           |
                 * .             . *     ---------------------------
                     *    .   *
                        F |                                  +
                          |+++++++++++++++++++++++++++++++++++
                          .
                     . *     *  .        ---------------------------
                 . *     IF      * .  T  |   QMIN=THSFDM(INDEXM)     |
               *      QMIN.GE.      *++++|                           |
               *THSFDM(INDEXM)      *    |                           |
                 * .             . *     ---------------------------
                     *    .   *
                        F |                                  +
                          |+++++++++++++++++++++++++++++++++++

             CALL FLBCDE(SUM(1),6,ANSO)

             CALL FLBCDE(SUM(2),6,ANSS)
                          |
                  +--------------+
                  +  DO LOOP TO  +
      ++++++++++++ STMT #    381 +
      +           + I=1,3        +
      +           +--------------+
      +                   |
      +      -----------------------------------------------------
 381 +++++++++| SUBT(I+2) =ANSO(I)                                 |
             -----------------------------------------------------
                          |
                  +--------------+
                  +  DO LOOP TO  +
      ++++++++++++ STMT #    382 +
      +           + I=1,3        +
      +           +--------------+
      +                   |
      +      -----------------------------------------------------
 382 +++++++++| SUBT(I+11) =ANSS(I)                                |
             | N     =1                                            |
             | L     =24                                           |
             -----------------------------------------------------
                          |
                  +--------------+
                  +  DO LOOP TO  +
      ++++++++++++ STMT #     53 +
      +           + I=1,MAXI     +
      +           +--------------+
      +                   |
      +      -----------------------------------------------------
      +      | QOUT(I) =0.0                                        |
      +      -----------------------------------------------------
      +                   |
  53 +++++++++CONTINUE
                          |
                  +--------------+
                  +  DO LOOP TO  +
      ++++++++++++ STMT #     57 +
      +           + I=1,MAXI     +
      +           +--------------+
      +                   |
      +           +--------------+
      +           +  DO LOOP TO  +
      ++++++++++++ STMT #     55 +
      +           + K=N,L        +
      +           +--------------+
      +                   |
      +      -----------------------------------------------------
      +      | QOUT(I) =Q(K)+QOUT(I)                               |
      +      -----------------------------------------------------
      +                   |
```

```
      +
55    ++++++++CCNTINUE
      +            |
      +       --------------------------------------------
      +       | QOUT(I) =QOUT(I)/24.0                    |
      +       | N       =K                               |
      +       | L       =N+23                            |
      +       --------------------------------------------
      +            |
      +
57    ++++++++CCNTINUE
                   |
              --------------------------------------------
              | APREC(3) =APREC(1)                       |
              --------------------------------------------
                   |
                 ------
                 |1000 |
                 ------


25            CCNTINUE
                   |
              --------------------------------------------
              | PEAK =PEAKS(1)                           |
              | QMIN =THSFDM(1)                          |
              --------------------------------------------
                   |
                +-----------+
                + DO LOOP TO +
      ++++++++++ STMT #   13 +
      +         + I=1,3      +
      +         +-----------+
      +            |
      +       --------------------------------------------
      +       | PEAK =AMAX1(PEAK,PEAKS(I))               |
      +       | QMIN =AMIN1(QMIN,THSFDM(I))              |
      +       --------------------------------------------
      +            |
13    ++++++++CCNTINUE
                   |
             * * IF * *      T     ------------------------------
           * MOD(SIND  *  *++++    |      IFLAG=1               |
          *EX,2).EQ.0     *        |                            |
             * *   * *             ------------------------------
               F                                               +
                   |++++++++++++++++++++++++++++++++++++++++++++
              --------------------------------------------
              | PEAK =AINT(PEAK+1)                       |
              | QMAP =PEAK                               |
              --------------------------------------------
                   |
                +-----------+
                + DO LOOP TO +
      ++++++++++ STMT #   17 +
      +         + I=1,3      +
      +         +-----------+
      +            |
      +         +-----------+
      +         + DO LOOP TO +
      ++++++++++ STMT #   19 +
      +         + K=1,SINDEX +
      +         +-----------+
      +            |
      +       --------------------------------------------
      +       | SQ(K)=THSFD(K,I)                         |
      +       --------------------------------------------
      +            |
19    ++++++++CCNTINUE
      +       CALL INTEG(IFLAG,H,SQ,SSUM,SINDEX)
      +            |
      +       --------------------------------------------
      +       | SUM(I) =SSUM/(26.9*AREA1)                |
      +       --------------------------------------------
      +            |
```

```
17 ++++++++CONTINUE
   | M     =2
   | KNT   =3
   | QMAX  =PEAKS(3)
   | PHRO  =PHRS(3)
   | SSUM  =SUM(3)

   DO LOOP TO
   STMT #    11
   I=1,3

   | PCHAR(I) =PCHARF(I)
   | PRCHAR(I) =PFCHAR(I)

11 ++++++++CONTINUE

   DO LOOP TO
   STMT #    33
   K=1,SINDEX

   | Q(K) =THSFD(K,1)

33 ++++++++CONTINUE

   DO LOOP TO
   STMT #     3
   I=1,18

   | SUBT(I) =FLOT(I)
   | FLDY(I) =FLDQ(I)

3 ++++++++CONTINUE
1000
        IF
     IPLOT.NE    T  ++++  GO TO 1001
     0
        F

   CALL PL360(NPPTS,TEMP,T,T(1),T(NQ),SUBT,Q,QMIN,QMAP,FLDY,
   TITLA,-214)

   CALL PL360(STPTS,TEMP,T,T(1),T(NQ),SUBT,SQ,QMIN,
   QMAP,FLDY,TITLA,226)

C

C SC4020 PLOTS

1001
        IF
     IPLOT.EQ    T  ++++  GO TO 391
     0
        F
```

```
CALL PL4020(1,1,PCHAR(1),SINDEX,T,Q,T(1),T(NQ),QMIN,QMAP,SUBT,
FLDY,TITLA,IERR)

CALL PL360(SINDEX,TEMP,T,T(1),T(NQ),SUBT,Q,QMIN,QMAP,FLDY,TITLA,
PRCHAR(1))

        IF
     IERR.EQ.          T
*1                          GO TO 185
        F

191     IF
     IERR.EQ.          T
*2                          GO TO 187
        F

     WRITE (6,715)

                        FORMAT   715 FORMAT (' OFF SCALE PLOT
                                 POINTS WERE ENCOUNTERED')

     120

187  WRITE (6,717)

                        FORMAT   717 FORMAT(' SUBROUTINES UNABLE
                                 TO CONSTRUCT READABLE GRIDS')

185   DO LOOP TO
      STMT #    21
      I=M,KNT

      DO LOOP TO
      STMT #    23
      K=1,SINDEX

   SQ(K)=THSFD(K,I)

23 CONTINUE

        IF
     CONOPT(1          T
*5).EQ.2                    PCHAR(1)=PCHAR(2)
        F
```

```
  +
  +               *
  +           *   *   *               -------------------------------
  +       . *     IF      * .   T    |                               |
  +     *      CONOPT(1           *++++|       PRCHAR(1)=PPCHAR(2)       |
  +     *5)   .FQ.2               *    |                               |
  +       * .                 . *     |                               |
  +           *   .   *               -------------------------------
  +               F |
  +                 |++++++++++++++++++++++++++++++++++++++++++++++++
  +                                                    +
  +      CALL PL4020(2,1,PCHAR(I),SINDEX,T,SQ,T(1),T(NQ),QMIN,
  +      QMAP,SUBT,FLDY,TITLA,IERR)
  +
  +      CALL PL360(SINDEX,TEMP,T,T(1),T(NQ),SUBT,SQ,QMIN,QMAP,FLDY,TITLA,
  +      PRCHAR(I))
  +
21 ++++++++CONTINUE
                  |
                  |                   -------------------------------
                  .                  |                               |
              *   *   *              |       GO TO 191               |
          . *     IF      * .   T    |                               |
        *      IERR.NE.           *++++|                               |
        *1                        *    |                               |
          * .                 . *     -------------------------------
              *   .   *
                  F |
                    |
                                    FORMAT   690 FORMAT(' ',18A4)
      C OBTAIN THE DIFFERENCE BETWEEN THE OBSERVED AND SIMULATED MAX FOR DAY

      C
                    |
               -------------------------------------------
391           | DIFFS=PEAKS(1)-QMAX                       |
              | DIFFSA =DIFFS                             |
               -------------------------------------------
                    |
                    |                 -------------------------------
              *   *   *              |                               |
          . *     IF      * .   T    |       DIFFSA=-DIFFS           |
        *      DIFFS.LT           *++++|                               |
        *.G.0                     *    |                               |
          * .                 . *     -------------------------------
              *   .   *                                  +
                  F |
                    |++++++++++++++++++++++++++++++++++++++++++++++
               -------------------------------------------
              | DIFFMP =(DIFFSA/QMAX)*100.0               |
               -------------------------------------------
                    |
      C

      C OBTAIN THE DIFFERENCE BETWEEN THE OBSERVED AND SIMULATED PEAKS
                    |
               -------------------------------------------
              | DIFFP=PHRS(1)-PHRO                        |
              | DIFFPA =DIFFP                             |
               -------------------------------------------
                    |
                    |                 -------------------------------
              *   *   *              |                               |
          . *     IF      * .   T    |       DIFFPA=-DIFFP           |
        *      DIFFP.LT           *++++|                               |
        *.0                       *    |                               |
          * .                 . *     -------------------------------
              *   .   *                                  +
                  F |
                    |++++++++++++++++++++++++++++++++++++++++++++++
               -------------------------------------------
              | DIFFPP =(DIFFPA/PHRO)*100                 |
               -------------------------------------------
                    |
      C

      C
```

```
C CCMPUTE DIFFERENCE AND %DIFFERENCE IN RUNOFF
                      |
-------------------------------------------------
| DIFFR=SUM(1)-SSUM                             |
-------------------------------------------------
                      |
-------------------------------------------------
| DIFFRA =DIFFR                                 |
-------------------------------------------------
                      |
                      *                      ---------------------------------
              . * *   *   * .               |      DIFFRA=-DIFFR            |
         . *         IF         * .    T     |                               |
        *        DIFFR.LT              *****|                               |
        *.0.0                          *     |                               |
           *  .      *   *     . *          ---------------------------------
                   F  |                                   +
                      |++++++++++++++++++++++++++++++++++++
-------------------------------------------------
| DIFFRP =(DIFFRA/SSUM)*100.0                   |
-------------------------------------------------
                      |
C

C

C      COMPUTE DIFFERENCE BETWEEN FORCASTED&WORSTCASE PRECIP
                      |
-------------------------------------------------
| DIFFPR =APREC(1)-APREC(3)                     |
| DIFPP=DIFFPR/APREC(1)*100                     |
-------------------------------------------------
                      |
      +--------------------------------------------------+
     + WRITE (6,700)                                     +
    +  (TITLE(I),I=1,18)                                +
   +                                                    +
  +--------------------------------------------------+
                      |
                      *                      ---------------------------------
              . * *   *   * .               |      GO TO 729                |
         . *         IF         * .    T     |                               |
        *        CONOPT(1              *****|                               |
        *5).EQ.2                       *     |                               |
           *  .      *   *     . *          ---------------------------------
                   F  |
                      |                       FORMAT   700 FORMAT('1',18A4)
                      |
      +--------------------------------------------------+
     + WRITE (6,710)                                     +
    +  MSBDIC                                           +
   +                                                    +
  +--------------------------------------------------+
                      |
                                           FORMAT   710 FORMAT(63X,'TABLE',1X
                                          ,'ONE',3X,'FORECAST RUN',/,64X,A4,10X,'WORST
    CASE',10X,'NO PRECIP',10X,'FORECAST',/)
                  +----------+
                  + DO LOOP TO   +
  ++++++++++++++++ STMT #     71 +
  +               + I=1,MAXI     +
  +               +----------+
  +                   |
  +    +--------------------------------------------------+
  +   + WRITE (6,711)                                     +
  +  +  MPDAY(I),(TMPREC(I,K),K=1,3),(TMOF(I,K),K=1,3),  +
  + +                                                    +
  + +--------------------------------------------------+
```

```
(TMIF(I,K),K=1,3),(TMBF(I,K),K=1,3),(STMROS(I,K),K=1,3)
                                    FORMAT  711 FORMAT(64X,I2,2X,'PRECIP'
                                    ,8X,F5.3,2(15X,F5.3),/,68X,'SUR R/O',7X,
F5.3,2(15X,F5.3),/,68X,'INT FL',8X,F5.3,2(15X,F5.3),/,68X,
'BASE FL',7X,F5.3,2(15X,F5.3),/,68X,'STM R/O',3X,F7.1,2(15X,F7.1),
//)

71 ++++++++CONTINUE

WRITE (6,700)
(TITLE(I),I=1,18)

WRITE (6,712)

                                    FORMAT  712 FORMAT(63X,'TABLE',1X,
                    'TWO',3X,'FORECAST RUN',/,43X,'WORST CASE',
10X,'NO PRECIP',10X,'FORECAST',10X,'DIFF   ',8X,'%DIFF',/)

WRITE (6,713)
(PEAKS(I),I=1,3),DIFFS,DIFFMP

                                    FORMAT  713 FORMAT(31X,'PEAK (CFS)'
//)                                 ,3X,F7.1,11X,F7.1,11X,F7.1,8X,F8.2,8X,F5.2,

WRITE(6,714)
(PHRS(I),I=1,3),DIFFP,DIFFPP

                                    FORMAT  714 FORMAT(31X,'PEAK (HR)',6X
                                    ,I3,15X,I3,15X,I3,12X,I3,10X,F5.2,//)

WRITE (6,719)
(SUM(I),I=1,3),DIFFR,DIFFRP

                                    FORMAT  719 FORMAT(31X,'R/O (IN)',9X,
//)                             F5.2,12X,F5.2,12X,F5.2,12X,F5.2,10X,F5.2,

WRITE(6,721)
(APREC(I),I=1,3),DIFFPR,DIFPP

                                    FORMAT  721 FORMAT(31X,'PRECIP (IN)'
F5.2)                               ,2X,F7.3,11X,F7.3,11X,F7.3,8X,F8.3,8X,

WRITE (6,700)
(TITLE(I),I=1,18)

WRITE(6,723)
MSBDIC
```

```
                          |                    FORMAT  723 FORMAT(63X,'TABLE',
                          |                    1X,'THREE',3X,'FORECAST RUN',/,64X,A4,10X,
           'WORST CASE',10X,'NO PRECIP',10X,'FORECAST',/)
                          |
                +-------------+
                + DO LOOP TO  +
      +++++++++++ STMT #   725 +
      +         + I=1,MAXI    +
      +         +-------------+
      +               |
      +      +------------------------------------------------+
      +     +  WRITE(6,727)                                  +
      +    +   MPDAY(I),(EMIFS(I,K),K=1,3),(EMUZS(I,K),K=1,3),+
      +   +                                                +
      +  +------------------------------------------------+
      +               |
      +      (EMLZS(I,K),K=1,3),(EMGWS(I,K),K=1,3),(EMBFNX(I,K),K=1,3),
      +      (EMSIAM(I,K),K=1,3),(EMUZC(I,K),K=1,3)
      FORMAT   727 FORMAT(64X,I2,2X,'IFS',11X,F5.2,2(15X,F5.2),/,68X,'UZS',11X,F5.2,
      +      2(15X,F5.2)/,68X,'LZS',11X,F5.2,2(15X,F5.2),/,68X,'GWS',11X,F5.2,
      +      2(15X,F5.2),/,68X,'BFNX',11X,F5.2,2(15X,F5.2),/,68X,'SIAM'10X,
      +      F5.2,2(15X,F5.2),/,68X,'UZC ',11X,F5.2,2(15X,F5.2),/)
      +
  725 ++++++++CONTINUE
                      |
                    |751  |

  729         CONTINUE
                      |
         +------------------------------------------------+
        +  WRITE (6,731)                                 +
       +   MSBOIC                                       +
      +                                                +
     +------------------------------------------------+
                      |
                                     FORMAT  731 FORMAT(63X,'TABLE',1X
        ,10X,'SIMULATED',//)     ,'ONE',10X,'PAST RUN',/,64X,A4,10X,'OBSERVED'
                      |
                +-------------+
                + DO LOOP TO  +
      +++++++++++ STMT #    85 +
      +         + I=1,MAXI    +
      +         +-------------+
      +               |
      +      +------------------------------------------------+
      +     +  WRITE(6,735)                                  +
      +    +   MPDAY(I),TMPREC(I,1),TMPREC(I,1),TMOF(I,1),TMIF+
      +   +(I,1),                                          +
      +  +------------------------------------------------+
      +               |
      +       TMBF(I,1),QOUT(I),STMROS(I,1)
                                         FORMAT  735 FORMAT(66X,I2,'PRECIP',8X
                                        ,F5.3,15X,F5.3,/,68X,'SUR R/O',27X,F5.3,
      +      /,68X,'INT FL',28X,F5.3,/,68X,'BASE FL',27X,F5.3,/,68X,'STM R/O',
      +      7X,F7.1,10X,F7.1,//)
      +
   85 ++++++++CONTINUE
                      |
         +------------------------------------------------+
        +  WRITE (6,700)                                 +
       +   (TITLE(I),I=1,18)                            +
      +                                                +
     +------------------------------------------------+
                      |
         +------------------------------------------------+
        +  WRITE (6,737)                                 +
       +                                                +
      +                                                +
     +------------------------------------------------+
                      |
```

```
                                             FORMAT  737 FORMAT(63X,'TABLE',1X,
                                      'TWO',10X,'PAST RUN',/,46X,'OBSERVED',2X,
     'SIMULATED',3X,'DIFF',3X,'% DIFF',/)

     WRITE (6,739)
     QMAX,PEAKS(1),DIFFS,DIFFMP

                                             FORMAT  739 FORMAT(11X,'PEAK (CFS)'
                                                 ,23X,2(F7.1,3X),F8.2,1X,F5.1,/)

     WRITE(6,741)
     PHRO,PHRS(1),DIFFP,DIFFPP

                                             FORMAT  741 FORMAT(11X,'PEAK (HR)
                                                 ',24X,I3,9X,I3,9X,I3,2X,F5.1,/)

     WRITE (6,743)
     SUM(1),SUM(2),DIFFR,DIFFRP

                                             FORMAT  743 FORMAT(11X,'R/O (IN)'
                                            ,27X,F5.2,5X,F5.2,6X,F5.2,1X,F5.1,/)

     WRITE (6,745)
     APREC(1),APREC(1)

                                             FORMAT  745 FORMAT(11X,'PREPCIP IN',
                                                  23X,F7.3,3X,F7.3)

     WRITE (6,700)
     (TITLE(I),I=1,18)

     WRITE (6,747)
     MSBDIC

                                             FORMAT  747 FORMAT(63X,'TABLE',
     D')                           1X,'THREE',2X,'PAST RUN',/,64X,A4,10X,'SIMULATE

             DO LOOP TO
             STMT #     87
             I=1,MAXI

     WRITE(6,749)
     MPDAY(I),EMIFS(I,1),EMUZS(I,1),EMLZS(I,1),EMGWS
     (I,1),

     EMBFNX(I,1),EMSIAM(I,1),EMUZC(I,1)
                                             FORMAT  749 FORMAT(66X,I2,2X,'IFS
                                       ',10X,F5.2,/,70X,'UZS ',10X,F5.2,/,70X,
     'LZS ',10X,F 5.2,/,70X,'GWS ',10X,F5.2,/,70X,'BFNX',10X,F5.2,/,
     70X,'SIAM',10X,F5.2,/,70X,'UZC ',10X,F5.2,//)

87   CONTINUE
```

```
751        CCNTINUE
777        CCNTINUE
           RETURN
      C
                                      FORMAT   603 FORMAT('OPOINT
                 |                 NUMBER',I5,' IS OUTSIDE PLOT LIMITS')
           ---------------------------------------
112        | L      =K-1                         |
           ---------------------------------------
                 |
               -------
               |113  |
               -------

END
```

```
SUBROUTINE INTEG(IFLAG,DT,Q,SUM,NQ)
C
      DIMENSION  Q(1)
      --------------------------------------------
      | SUM   =0.0                               |
      --------------------------------------------
                  |
               . * . * .                ------------------------------
           . *    IF     * .   T       |                              |
         *   IFLAG.EQ        *++++++++|     GO TO 80                 |
         *.2                 *         |                              |
           * .            . *          ------------------------------
               * . * . *
                 F |
      --------------------------------------------
      | LU    =NQ-1                              |
      --------------------------------------------
                  |
             +--------------+
            +  DO LOOP TO    +
  ++++++++++++ STMT #     40 +
  +         +  I=2,LU        +
  +          +--------------+
  +               |
  +               |
  +       --------------------------------------------
40 ++++++++| SUM   =SUM+2.*Q(I)                      |
          | SUM   =(SUM+Q(1)+Q(NQ))*DT*.5           |
          --------------------------------------------
                  |
      RETURN
C
                  |
          --------------------------------------------
80        | LU    =NQ-3                              |
          --------------------------------------------
                  |
             +--------------+
            +  DO LOOP TO    +
  ++++++++++++ STMT #    100 +
  +         +  I=2,LU,2      +
  +          +--------------+
  +               |
  +               |
  +       --------------------------------------------
100 +++++++| SUM   =SUM+4.*Q(I)+2.*Q(I+1)            |
          | SUM   =DT/3.*(Q(1)+SUM+4.*Q(NQ-1)       |
          |          +Q(NQ))                         |
          --------------------------------------------
                  |
      RETURN
END
```

```
      SUBROUTINE LABELS(BCDX,BCDY,IDF,IERR)

      CCMMCN/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
            NULLV  ,IGDSC  ,IPINOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
            IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
            VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
            XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
            YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
            AY     ,BY     ,YLPOS  ,
            LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
            FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      LCGICAL*1 GSPIN, DUM(2)
      INTEGER*4 PTLID(3)/'PLOT',' NO ','    '/
      INTEGER*4 EDVAL(3),ZERO(3)/'    ','  0 ','    '/
      INTEGER*4 IDF(18),BCDX(18),BCDY(18),BCDY2(72)
      REAL*8 DYDVAL,WORK(36)
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      LCGICAL*1 FA,FB,FC,GMODE1
      LCGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      EQUIVALENCE (RWORK1,RN),(ISYM,IRNO)
C     CREATE PLOT TITLE LABEL
                        I
      ------------------------------------------------
      | ICNT  =IFIX(XSIZE/HSPB)+5                     |
      | RN    =XLL-5.*HSPB                            |
      ------------------------------------------------
                        I
      CALL PTEXT(IGDSC,IDF,ICNT,NULLV,NULLV,1,RN,YUR+5.*VSPB)
C     CREATE X AXIS LABEL ELEMENTS

      CALL PTEXT(IGDSC,BCDX,ICNT,NULLV,NULLV,1,RN,YLL-3.5*VSPB)
C     CREATE Y AXIS LABEL ELEMENTS

       CALL PLOTL(BCDY,BCDY2,72,IRNO)
                        I
      ------------------------------------------------
      | RN    =YUR+6.*VSPB                            |
      | IRNO  =MIN0(IRNO,IFIX(YSIZE/VSPB)             |
      |        +5)                                    |
      ------------------------------------------------
                        I
                  +-------------+
                  +  DO LOOP TO  +
      +++++++++++ +  STMT #    40 +
      +           +  I = 1,IRNO   +
      +           +-------------+
      +                 I
      +                 I
      +   ------------------------------------------------
      +   | RN    =RN-VSPB                              |
      +   ------------------------------------------------
      +                 I
      +    CALL PTEXT(IGDSC,BCDY2(I),1,NULLV,NULLV,1,0.0,RN)
      +
   40 ++++++++ CONTINUE
                        I
                     .  *  .
                . *     IF    * .    T    ------------------------
              *    IERR  .EQ       *++++|   GO TO 70            |
              *. 2                 *    |                       |
                 *  .        .  *       ------------------------
                     *  *
                     F  I
```

```
C   CCNVERT Y DIVISION VALUES TO EBCDIC FORMAT AND CREATE

C   THE TEXT ELEMENTS NECESSARY FOR DISPLAYING
                  |
              +-------------+
              + DO LOOP TO  +
  +++++++++++ + STMT #   50 +
  +           + I=1,NYD     +
  +           +-------------+
  +                 |
  +        ------------------------------------------------
  +        | DYDVAL =YDVAL(I)                             |
  +        ------------------------------------------------
  +                 |
  +        CALL TC3(DYDVAL,11,EDVAL)
50 ++++++++CALL PTEXT(IGDSC,EDVAL,11,NULLV,NULLV,1,XLL-13.*HSPB,YTIC(I))
C   CCNVERT X DIVISION VALUES TO EBCDIC FORMAT AND CREATE

C   THE TEXT ELEMENTS NECESSARY FOR DISPLAYING
                  |
              +-------------+
              + DO LOOP TO  +
  +++++++++++ + STMT #   60 +
  +           + I=1,NXD     +
  +           +-------------+
  +                 |
  +        ------------------------------------------------
  +        | DYDVAL =XDVAL(I)                             |
  +        ------------------------------------------------
  +                 |
  +        CALL TC3(DYDVAL,9,EDVAL)
  +                 |               ------------------------------
  +            * * * *              |                            |
  +         *    IF     *     T     |     GO TO 60               |
  +       *   XDVAL(I)    *+++++    |                            |
  +       *  .NE. 0.0     *         |                            |
  +         *           *           ------------------------------
  +            * * * *
  +               F
  +               |
  +        ------------------------------------------------
  +        | EDVAL(1) =ZERO(1)                            |
  +        | EDVAL(2) =ZERO(2)                            |
  +        | EDVAL(3) =ZERO(3)                            |
  +        ------------------------------------------------
  +                 |
60 ++++++++CALL PTEXT(IGDSC,EDVAL,9,NULLV,NULLV,1,XTIC(I)-6.*HSPB,
                YLL-1.5*VSPB)
                  |
C   CREATE PLOT ID

70         CCNTINUE
                  |
           ------------------------------------------------
           | RN   =XLL+(XSIZE/2.-3.*HSPB)                 |
           ------------------------------------------------
                  |
           RETURN

END
```

```
SUBROUTINE PL4020(NPLOT,MODE,NCHAR,NP,X,Y,XMIN,XMAX,
                          YMIN,YMAX,BCDX,BCDY,IDF,IERR)
C   THIS IS THE MAIN ROUTINE FOR GENERATING 2250 PLOTS
C
C   INPUT PARAMETER DEFINITION:
C
C     NPLOT      - CONTROLS NUMBER OF CURVES PER DISPLAY
C                  (NPLOT=1 START NEW DISPLAY)
C                  (NPLOT=2 SUPERIMPOSE CURVE ON PREVIOUS DISPLAY)
C     MODE       - TYPE OF GRID
C                  = 1 X-LINEAR AND Y-LINEAR
C                  = 2 X-LOG    AND Y-LINEAR
C                  = 3 X-LINEAR AND Y-LOG
C                  = 4 X-LOG    AND Y-LOG
C     NCHAR      - SELECTS PLOT SYMBOL TO BE USED
C     NP         - NUMBER OF POINTS TO BE PLOTTED
C     X,Y        - NAMES OF ARRAYS CONTAINING THE X AND Y COORDINATES
C     XMIN,XMAX - MINIMUM AND MAXIMUM VALUES FOR X COORDINATES
C     YMIN,YMAX - MINIMUM AND MAXIMUM VALUES FOR Y COORDINATES
C     BCDX,BCDY - CHARACTER LABELS FOR X AND Y AXES (72 CHAR MAX)
C     IDF        - NAME OF 72 CHARACTER HEADING FOR EACH DISPLAY
C     IERR       - ERROR INDICATOR.  NOT SET BY THIS ROUTINE
C                  SINCE ERROR MESSAGES ARE DISPLAYED ON THE
C                  2250 SCREEN.  STANDARD MEANINGS ARE GIVEN BELOW.
C                   = 1 NORMAL RETURN
C                   = 2 UNABLE TO CONSTRUCT READABLE GRAPH
C                   = 3 OFF-SCALE PLOT POINTS ENCOUNTERED
C
      COMMON/GSPD/WORK  ,GSP1  ,UNITN ,I2250 ,IATN  ,OAREA ,
     NULLV ,IGDSC ,IPTNOW,MAXPT ,IPT   ,ISYM  ,RWORK1,
     IGDS4 ,IGDS5 ,IGDS6 ,IDSP  ,RU    ,HSPB  ,HSPL  ,
     VSPB  ,VSPL  ,NOUT  ,NXD   ,NYD   ,XSIZE ,YSIZE ,
     XLL   ,YLL   ,XUR   ,YUR   ,XTIC  ,YTIC  ,XDVAL ,
     YDVAL ,IKEY1 ,IKEY2 ,IKEY3 ,IKEY4 ,AX    ,BX    ,
     AY    ,BY    ,YLPOS ,
     LOGXSW,LOGYSW,NPTF  ,NRGF  ,ERRHD ,FA    ,FB    ,
     FC    ,GMODE1,GSPIN ,DUM   ,IRECAL
```

```
      LOGICAL*1 GSPIN, DUM(2)
      REAL*8 ERR1(6)/'NO POINT','S SPECIF','IED FOR ','1 OR MOR',
     'E CURVES','        '/
      REAL*8 ERR2(6)/'UNABLE T','O CONSTR','UCT READ','ABLE GRA',
     'PH      ','        '/
      REAL*8 PLTID(2)/'<<< ERRO','RS >>>  '/
      DIMENSION A(1)
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      REAL*4 LGXMAX,LGXMIN,LGYMAX,LGYMIN
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      LOGICAL*1 FA,FB,FC,GMODE1
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      DIMENSION X(NP),Y(NP)
      REAL*8 BCDX(9),BCDY(9),IDF(9),WORK(36)
             |
      ----------------------------------------
      | IERR =1                              |
      ----------------------------------------
             |
      GO TO(10,200),NPLOT
10    CONTINUE
      CALL SHIFT1
             |
      ----------------------------------------
      | LOGXSW =.FALSE.                      |
      | LOGYSW =.FALSE.                      |
      | FA    =.FALSE.                       |
      | IPTNOW =0                            |
      | ERRHD=.TRUE.                         |
      | NPTF =.FALSE.                        |
      | NRGF =.FALSE.                        |
      ----------------------------------------
             |
   +----------------------------------------------+
  +  WRITE(NOUT,1020)                            +
 +  IDSP,(IDF(I),I=1,9),NPLOT,MODE,NCHAR,NP,     +
+----------------------------------------------+
             |
         XMIN,XMAX,YMIN,YMAX
13    CALL SGRID
             |
        . * . * .               ------------------------
     *     IF      *  .  T      |                      |
   *    XMAX .LE       *+++++  |     GO TO 100        |
   *. XMIN .OR. YMAX .L*        |                      |
   *E. YMIN            *        |                      |
        * .  * . *              ------------------------
           F |
      GO TO(20,30,40,50),MODE
C
C     MODE = 1    X-LINEAR Y-LINEAR
C
20    CONTINUE
             |
      ----------------------------------------
      | AX    =XSIZE/(XMAX-XMIN)             |
      | BX    =XLL-AX*XMIN                   |
      | AY    =YSIZE/(YMAX-YMIN)             |
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
| BY    =YLL-AY*YMIN                        |
| XINC  =(XMAX-XMIN)/(FLOAT(NXD)-1.0)       |
| YINC  =(YMAX-YMIN)/(FLOAT(NYD)-1.0)       |
-------------------------------------------
          |
       --------
       |60    |
       --------

C
C      MODE = 2    X-LOG     Y-LINEAR
C
30       CONTINUE
            |
         . * * .                    -----------------------------
      . *   IF    * .  T           |      GO TO 100              |
     *     XMIN .LE    *+++++      |                             |
     *. 0.0            *           |                             |
        *  .      . *               -----------------------------
            * . *
            F |
-------------------------------------------
| LGXMIN =ALOG10(XMIN)                      |
| LGXMAX =ALOG10(XMAX)                      |
| LOGXSW =.TRUE.                            |
| AX     =XSIZE/(LGXMAX-LGXMIN)             |
| BX     =XLL-AX*LGXMIN                     |
| AY     =YSIZE/(YMAX-YMIN)                 |
| BY     =YLL-AY*YMIN                       |
| XINC   =(XMAX/XMIN)**(1.0/(FLOAT(NXD)     |
|         -1.0))                            |
-------------------------------------------
-------------------------------------------
| YINC =(YMAX-YMIN)/(FLOAT(NYD)-1.0)        |
-------------------------------------------
          |
       --------
       |60    |
       --------

C
C      MODE = 3    X-LINEAR Y-LOG
C
40       CONTINUE
            |
         . * * .                    -----------------------------
      . *   IF    * .  T           |      GO TO 100              |
     *     YMIN .LE    *+++++      |                             |
     *. 0.0            *           |                             |
        *  .      . *               -----------------------------
            * . *
            F |
-------------------------------------------
| LGYMAX =ALOG10(YMAX)                      |
| LGYMIN =ALOG10(YMIN)                      |
| LOGYSW =.TRUE.                            |
| AX     =XSIZE/(XMAX-XMIN)                 |
| BX     =XLL-AX*XMIN                       |
| AY     =YSIZE/(LGYMAX-LGYMIN)             |
| BY     =YLL-AY*LGYMIN                     |
| XINC   =(XMAX-XMIN)/(FLOAT(NXD)-1.0)      |
| YINC   =(YMAX/YMIN)**(1.0/(FLOAT(NYD)     |
|         -1.0))                            |
-------------------------------------------
```

```
                 |
             ---------
             |60     |
             ---------

C
C   MODE = 4    X-LOG     Y-LOG
C
50          CONTINUE
                 |
           . * * .                  ------------------------------
       . *   IF    * .    T        |    GO TO 100                 |
     *   XMIN .LE       *++++++    |                              |
     *. 0.0             *          |                              |
       * .         . *              ------------------------------
           * . *
           F |
             |
           . * * .                  ------------------------------
       . *   IF    * .    T        |    GO TO 100                 |
     *   YMIN .LE       *++++++    |                              |
     *. 0               *          |                              |
       * .         . *              ------------------------------
           * . *
           F |
             |
    --------------------------------------------
   | LGXMIN =ALOG10(XMIN)                       |
   | LGXMAX =ALOG10(XMAX)                       |
   | LGYMAX =ALOG10(YMAX)                       |
   | LGYMIN =ALOG10(YMIN)                       |
   | LOGXSW =.TRUE.                             |
   | LOGYSW =.TRUE.                             |
   | AX     =XSIZE/(LGXMAX-LGXMIN)              |
   | BX     =XLL-AX*LGXMIN                      |
   | AY     =YSIZE/(LGYMAX-LGYMIN)              |
   | BY     =YLL-AY*LGYMIN                      |
   | XINC =(XMAX/XMIN)**(1.0/(FLOAT(NXD)        |
   |       -1.0))                               |
   | YINC =(YMAX/YMIN)**(1.0/(FLOAT(NYD)        |
   |       -1.0))                               |
    --------------------------------------------
             |
C
60          CONTINUE
             |
    --------------------------------------------
   | XDVAL(1) =XMIN                             |
   | YDVAL(1) =YMIN                             |
    --------------------------------------------
             |
       +-------------+
       + DO LOOP TO  +
+++++++++ STMT #   72 +
+      + I=2,NXD     +
+      +-------------+
+            |
+          . * * .                  ------------------------------
+      . *   IF    * .    T        |    GO TO 70                  |
+    *   .NOT.LOG       *+++++     |                              |
+    *XSW               *          |                              |
+      * .         . *              ------------------------------
+          * . *
+          F |
+            |
+   --------------------------------------------
+  | XDVAL(I) =XDVAL(I-1)*XINC                  |
+   --------------------------------------------
+            |
```

```
              |
           |72    |
           --------

              |
         ------------------------------------------
70 +     | XDVAL(I)  =XDVAL(I-1)+XINC            |
         ------------------------------------------
              |
72 ++++++++CCNTINUE
              |
        +--------------+
        + DO LOOP TO   +
+++++++++ STMT #    82 +
+       + I=2,NYD      +
+       +--------------+
+             |
+             |                  ------------------------------
+         . * . * .              |                            |
+       *      IF      *   T     |   GO TO 80                 |
+      *   .NOT.LCG     *+++++   |                            |
+      *YSW            *         |                            |
+         . * . * .              ------------------------------
+            F |
+              |
+        ------------------------------------------
+        | YDVAL(I)  =YDVAL(I-1)*YINC            |
+        ------------------------------------------
+              |
+          --------
+          |82    |
+          --------
+
+              |
+        ------------------------------------------
80 +     | YDVAL(I)  =YDVAL(I-1)+YINC            |
+        ------------------------------------------
+              |
82 ++++++++CCNTINUE
              |
          . * . * .
        *     IF     *
        *    (NP)    *
      |   . * . * .    |
    - |      0 |     + |
 --------  --------  --------
 |210   |  |210   |  |90    |
 --------  --------  --------

90        CALL POINTS(X,Y,NP,NCHAR,IERR,XMIN,XMAX,YMIN,YMAX,PLTID)
81        CALL LABELS(BCDX,BCDY,IDF,IERR)
      C   ENTER PLOT TITLE IN DICTIONARY

          CALL DICTRY(IDF)
              |
          --------
          |205   |
          --------


      C

100       CCNTINUE
          CALL PTEXT(IGDSC,ERR2,48,NULLV,NULLV,1,0.0,YLL-10.*VSPB)
```

```
           WRITE(NOUT,1030)
           (ERR2(I),I=1,6)

103        NRGF =.TRUE.
           IERR =2

           |211 |

     C
     C   CONTROL PASSES TO STMT 200 WHEN NPLOT EQ 2
     C
200        CONTINUE

           WRITE(NOUT,1010)
           NPLOT,NCHAR,NP

              IF
           NRGF              T   GO TO 230
             F

              IF
           NP .LE.           T   GO TO 210
           0
             F

           CALL POINTS(X,Y,NP,NCHAR,IERR,XMIN,XMAX,YMIN,YMAX,PLTID)
205        CONTINUE

              IF
           GMODE1            T   GO TO 230
             F

           CALL EXEC(IGOSC)
           CALL ATTEN

           |230 |

210        WRITE(NOUT,1030)
           (ERR1(I),I=1,6)
```

```
                    |
              . * . * .
         * . *    IF    * .   T    ---------------------------------
       *      NPTF          *****|   GO TO 212                      |
         *. .            . *       ---------------------------------
              * . *
             F  |
      ------------------------------------------
      | NPTF =.TRUE.                           |
      ------------------------------------------
                |
      CALL PTEXT(IGDSC,ERR1,48,NULLV,NULLV,1,0.0,YLL-8.5*VSPB)
                |
211             .                  ---------------------------------
              . * . * .
         * . *    IF    * .   T  |   CALL PTEXT(IGDSC,PLTID,1        |
       *      ERRHD         *****|  4,NULLV,NULLV,1,0.,YLL-5.5       |
         * .             . *       ---------------------------------
              * . *                                   +
             F  |+++++++++++++++++++++++++++++++++++++
      ------------------------------------------
      | ERRHD=.FALSE.                          |
      ------------------------------------------
                |
212   GO TO(81,230),NPLOT
230   CCNTINUE
                |
      ------------------------------------------
      | IERR =1                                |
      ------------------------------------------
                |
      RETURN
   C
   C  DUMMY ENTRY FOR TX4020
   C
      ENTRY TX4020(NPLOT,A,NCHAR,NLINE,NCOL,IER)
                |
      +--------------------------------------------+
     +  WRITE(NOUT,2000)                          +
    +                                            +
   +                                            +
  +--------------------------------------------+
                |
RETURN
      FORMAT 1010 FORMAT(' PL4020:  NPLOT=',I1,'     NCHAR=',I2,'     NP=',I6)
      FORMAT 1020 FORMAT(' PL4020:  PLOT=',I5,3X,9A8/10X,'NPLOT=',I1,'    MODE=',
           I1,'    NCHAR=',I2,'    NP=',I5,'    XMIN=',E12.4,
           '    XMAX=',E12.4,'    YMIN=',E12.4,'    YMAX=',E12.4)
                                  FORMAT 1030 FORMAT(' ***',6A8)
                                  FORMAT 2000 FORMAT(' **** TX4020
                                       ENTRY NOT IMPLEMENTED')
END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE PLCTID(IDF)
      COMMCN/GSPD/WORK  ,GSP1  ,UNITN  ,I2250 ,IATN  ,OAREA ,
     *       NULLV  ,IGDSC ,IPTNOW,MAXPT ,IPT   ,ISYM  ,RWORK1,
     *       IGDS4  ,IGDS5 ,IGDS6 ,IDSP  ,RU    ,HSPB  ,HSPL  ,
     *       VSPB   ,VSPL  ,NOUT  ,NXD   ,NYD   ,XSIZE ,YSIZE ,
     *       XLL    ,YLL   ,XUR   ,YUR   ,XTIC  ,YTIC  ,XDVAL ,
     *       YDVAL  ,IKEY1 ,IKEY2 ,IKEY3 ,IKEY4 ,AX    ,BX    ,
     *       AY     ,BY    ,YLPOS ,
     *       LOGXSW,LOGYSW,NPTF  ,NRGF  ,ERRHD ,FA    ,FB    ,
     *       FC     ,GMODE1,GSPIN ,DUM   ,IRECAL
      INTEGER*4 STAR(19)/19*'*****'/
      LOGICAL*1 GSPIN, DUM(2)
      INTEGER*4 GSP1,UNITN,OAREA,NULLV(1)
      REAL*8 IDF(9),WORK(36)
      REAL*4 TBLCCN(5)/'TABL','E OF',' CON','TENT','S   '/
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6),GLAB(16)
      LCGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LCGICAL*1 FA,FB,FC,GMODE1
      ECUIVALENCE  (WORK(11),GLAB(1))
      EQUIVALENCE (RWORK1,Y,X,XINC,YINC)

          11111

      WRITE(NOUT,1000)
      (IDF(I),I=1,9)

C     INITIALIZE GRAPHIC SERVICES
      CALL INGSP(GSP1,NULLV)
C     SPECIFY SUBROUTINE LINK/LOAD STATUS
      CALL SPEC(GSP1,1,30,37,38,40)
      CALL SPEC(GSP1,2,17,-19,53,-56)
C     INITIALIZE THE 2250
      CALL INDEV(GSP1,UNITN,I2250)
C     INITIALIZE GRAPHIC DATA SETS
      CALL INGDS(I2250,IGDSC,OAREA)
      CALL INGDS(I2250,IGDS5,128)
      CALL INGDS(I2250,IGDS6,512)
C     SET NO SCISSIORING FOR ALL DATA SETS
      CALL SSCIS(IGDSC,3)
      CALL SSCIS(IGDS5,3)
      CALL SSCIS(IGDS6,3)
11111 CCNTINUE
C     COMPUTE X AND Y TIC MARK COORDINATES AND ADJUST GRID SIZE
C     PARAMETERS FOR TRUNCATICN ERRORS
```

```
              |
   ----------------------------------------
   | XINC =(XUR-XLL)/(FLOAT(NXD)-1.0)     |
   | XTIC(1) =XLL                         |
   ----------------------------------------
              |
        +-------------+
        + DO LOOP TO  +
+++++++++++ STMT #   10 +
+       + I=2,NXD     +
+       +-------------+
+             |
+    ----------------------------------------
10 ++++++++| XTIC(I) =XTIC(I-1)+XINC          |
   | XUR   =XTIC(NXD)                     |
   | XSIZE=XUR-XLL                        |
   | YSIZE=0.7*XSIZE                      |
   | YLL   =YUR-YSIZE                     |
   | YINC =YSIZE/(FLOAT(NYD)-1.0)         |
   | YTIC(1) =YLL                         |
   ----------------------------------------
              |
        +-------------+
        + DO LOOP TO  +
+++++++++++ STMT #   20 +
+       + I=2,NYD     +
+       +-------------+
+             |
+    ----------------------------------------
20 ++++++++| YTIC(I) =YTIC(I-1)+YINC          |
   | YUR   =YTIC(NYD)                     |
   | YSIZE=YUR-YLL                        |
   ----------------------------------------
              |
C    CREATE TEXT ELEMENTS FOR LIGHT PEN OPTION SELECTION IN IGDS5.
              |
   ----------------------------------------
   | X     =(RU-21.*HSPB)-1.0             |
   ----------------------------------------
              |
      CALL STPOS(IGDS5,X,4.5*VSPB)
      CALL PTEXT(IGDS5,GLAB,20,NULLV,IKEY1,2)
      CALL PTEXT(IGDS5,GLAB(6),20,NULLV,IKEY3,2,X,1.5*VSPB)
      CALL PTEXT(IGDS5,GLAB(11),12,NULLV,IKEY2,2,X,3.*VSPB)
      CALL EXEC(IGDS5)
      CALL INCL(IGDS5)
C    CREATE DICTIONARY STATIC ELEMENTS IN IGDS6

      CALL SCHAM(IGDS6,2)
      CALL STPOS(IGDS6,1375.,4095.)
      CALL PTEXT(IGDS6,TBLCON(1),17,-1,NULLV,2)
      CALL PTEXT(IGDS6,GLAB(11),12,-2,NULLV,2,3000.,0.)
      CALL PTEXT(IGDS6,GLAB(14),12,-3,NULLV,2,3000.,1.5*VSPL)
      CALL EXEC(IGDS6)
C    CREATE ATTENTION LEVEL FOR 2250

      CALL CRATL(12250,IATN)
C    DESIGNATE THAT BOTH CHARACTER CODE AND COORDINATES OF CHARACER

C    DETECTED BY THE LIGHT PEN ARE TO BE RETURNED BY RQATN SUBROUTINE

      CALL MLPEO(IATN,2,4,1)
C    SET L.P. ATTENTIONS AND LIGHT F.K. 0
```

```
          CALL SLPAT(IGDS5,1)
          CALL SLPAT(IGDS6,1)
          CALL MLITS(I2250,3)
      C   GENERATE ID FRAME IN IGDSC
          CALL STPOS(IGDSC,0.,2290.)
          CALL PTEXT(IGDSC,STAR,74,NULLV,NULLV)
          CALL PTEXT(IGDSC,STAR,74,NULLV,NULLV,1,0.0,1810.)
          CALL PTEXT(IGDSC,IDF,72,NULLV,NULLV,1,56.,2050.)
      C   CREATE VERTICAL BOX OUTLINE ELEMENTS
                          I
          -----------------------------------------------
          I Y     =1810.                                I
          -----------------------------------------------
                          I
            +---------------+
            +  DO LOOP TO   +
   ++++++++++  STMT #     1 +
   +        +  I=1,5        +
   +        +---------------+
   +                      I
   +      -----------------------------------------------
   +      I Y     =Y+80.                                I
   +      -----------------------------------------------
   +                      I
   +      CALL PTEXT(IGDSC,STAR,1,NULLV,NULLV,1,0.0,Y)
   +
 1 ++++++++CALL PTEXT(IGDSC,STAR,1,NULLV,NULLV,1,4088.,Y)
                          I
      C   PLACE ID FRAME TITLE IN PLOT DICTIONARY
          CALL DICTRY(IDF)                              FORMAT 1000 FORMAT('1',T24,
                                                               '<<< ',9A8,' >>>')
          RETURN
    END
```

```
SUBROUTINE PLTEND(IDUM)

      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     AY     ,BY     ,YLPOS  ,
     LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL,IFLAG

      LOGICAL*1 GSPIN, DUM(2)

      REAL*8 END(1)/'END      '/,WORK(36)

      INTEGER*4 GSP1,UNITN,NULLV(1),OAREA

      LOGICAL*1 FA,FB,FC,GMODE1

      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD

      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)

      | IRECAL =1                                      |

      CALL SHIFT1

      CALL PTEXT(IGDSC,END,4,NULLV,NULLV,1,1879.,2047.)

      CALL EXEC(IGDSC)

      | IFLAG=0                                        |
      | YLPOS=3796.0                                   |

      CALL  TMDEV(I2250)

      CALL TMGSP(GSP1)
                                     FORMAT   100 FORMAT(' ***END OF
                                                  GRAPHIC PORTION')
      RETURN

END
```

ROUTINE TO PRINT PL360 GRAPH

```
      C
      C      WRITTEN BY C G HOOKS          IBM HUNTSVILLE
      C
             SUBROUTINE PR360 (ARRAY)
      C
             DIMENSION ARRAY (1540)
             EQUIVALENCE (IEST,A)
             INTEGER *4  TEST /'DAYS'/

             | A     =ARRAY(1490)
             | A1    =ARRAY(1491)
             | A2    =ARRAY(1492)
             | A3    =ARRAY(1493)
             | A4    =ARRAY(1494)
      C
      C      TEST FOR JUST PRINTING THE FIELD OF ARRAY WITHOUT SCALES

             | IJ    =1515
             | JI    =1520

                    IF
           (ARRAY(IJ)-ARRAY(JI))
          -          0          +
         |41 |     |42 |      |41 |

   42        CONTINUE

             WRITE (6, 43)
             ARRAY
                                          FORMAT   43 FORMAT (28A4)
             RETURN
      C
   41        CONTINUE

             WRITE (6,6)
             (ARRAY(I), I = 1,56)
                                          FORMAT    6 FORMAT (2A4, 8X, 26A4)

             WRITE (6,9)
```

```
                                          FORMAT     9 FORMAT (16X,1H+, 5(19H-------------------, 1H+) )
            |
   +------------------------------------------------+
  +  WRITE (6,7)                                   +
 +   (ARRAY(I), I = 57,1456)                      +
+------------------------------------------------+
            |
                                          FORMAT     7 FORMAT(A4,F11.2,1H+,25A4,A1,1H+ / 2A4,7X,1H|,25A4,A1,1H| /
          2A4,7X,1H|,25A4,A1,1H| / 2A4,7X,1H|,25A4,A1,1H| /
          2A4,7X,1H|,25A4,A1,1H| / 2A4,7X,1H|,25A4,A1,1H| /
          2A4,7X,1H|,25A4,A1,1H| / 2A4,7X,1H|,25A4,A1,1H| /
          2A4,7X,1H|,25A4,A1,1H| / 2A4,7X,1H|,25A4,A1,1H| )
            |
   +------------------------------------------------+
  +  WRITE (6,5)                                   +
 +   (ARRAY(I), I = 1457,1484)                    +
+------------------------------------------------+
            |
                                          FORMAT     5 FORMAT(A4,F11.2,1H+,25A4,A1,1H+ )
            |
            *                        ------------------------
       *  .  * IF  *  *  .   T      |   GO TO 50            |
   *  .  TEST  .EQ            *++++++|                       |
   *.  TEST                   *     |                       |
       *  .   *   *  .  *            ------------------------
            F
            |
   +------------------------------------------------+
  +  WRITE (6,8)                                   +
 +   (ARRAY(I), I = 1515,1520)                    +
+------------------------------------------------+
            |
                                          FORMAT     8 FORMAT (16X,       1H+, 5(19H-------------------,1H+) /
      9X, F11.0,  5(9X,F11.0) /)
            |
   +------------------------------------------------+
  +  WRITE (6,6)                                   +
 +   (ARRAY(I), I = 1485,1512)                    +
+------------------------------------------------+
            |

       RETURN

50     CONTINUE
```

```
WRITE(6,11)

WRITE(6,10)

FORMAT   11 FORMAT(16X,'+OOOOOOONNNNNNNNNDDD+DDDDJJJJ'JJJJFFFFFF+',
'FMMMMMMMMAAAAAAAAMM+MMMMMMJJJJJJJJJJJJ+',
'JJAAAAAAAASSSSSSSSS+')
FORMAT   10 FORMAT(19X,'OCT.    NOV.     DEC.    JAN.     FEB.     ',
'.MARCH    APRIL    MAY      JUNE     JULY    ',
'. AUG.    SEPT.   ')

RETURN

END
```

```
SUBROUTINE POINTS(X,Y,NP,NCHAR,IERR,XMIN,XMAX,YMIN,YMAX,PLTID)

         COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
              NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
              IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
              VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
              XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
              YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
              AY     ,BY     ,YLPOS  ,
              LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
              FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL

         LOGICAL*1 GSPIN, DUM(2)

         REAL*8 ERR3(6)/'OFF SCAL','E POINTS',' ENCOUNT','ERED    ',
              2*'        '/,WORK(36),PLTID(2)

         LOGICAL*1 FA,FB,FC,GMODE1

         LOGICAL*1 LOGXSW,LOGYSW,OFSC,NPTF,NRGF,ERRHD

         INTEGER*4 UNITN,OAREA,NULLV(1),GSP1

         REAL*4 XPOS(36),YPOS(36),XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)

         LOGICAL*1 OFF

         DIMENSION X(NP),Y(NP)

         | ISAVE=NP
         | OFSC =.FALSE.

              IF FA  --T-->  | GO TO 102 |
                 F

         | IPTNOW =IPTNOW+NP

              IF (IPTNOW-MAXPT)
          -          0          +
         |1 |      |1 |      |7 |

7        CONTINUE

         | NP    =NP-(IPTNOW-MAXPT)
         | FA    =.TRUE.

         WRITE(NOUT,2000)
         MAXPT

              IF (NP)
          -          0          +
         |102 |    |102 |    |1 |
```

```
C    DETERMINE PLOT SYMBOL TO BE USED.  A VALUE OF 0 IS RETURNED
C    IF THE SELECTED PLOT SYMBOL IS 'POINTS' OR A NON-DISPLAYABLE
C    SYMBOL.  IN THE LATTER CASE 'POINTS' IS THE DEFAULT.
1          CONTINUE
           CALL PLOTSM(NCHAR,ISYM)
                     |
                .  *  .  *  .              ---------------------------------
            . *      IF       * .   T     |  GO TO 5                        |
           *    ISYM .EQ          *+++++  |                                 |
           *. 0                  *        |                                 |
              *  .   *   .  *              ---------------------------------
                  F  |
           ASSIGN 2 TO M
                     |
                +-------------+
                + DO LOOP TO  +
    ++++++++++++ STMT #     3 +
    +           + K=1,NP      +
    +           +-------------+
    +                |
    +        ------------------------------------------
    +        | OFF  =.FALSE.                           |
    +        ------------------------------------------
    +                |
    +           .  *  .  *  .              ---------------------------------
    +       . *      IF       * .   T     |  XPOS(1)=AX*ALOG10(X(K))        |
    +      *   LOGXSW           *+++++    |  + BX                           |
    +         *  .   *   .  *              ---------------------------------
    +             F  |                                   +
    +                |+++++++++++++++++++++++++++++++++++++
    +           .  *  .  *  .              ---------------------------------
    +       . *      IF       * .   T     |  XPOS(1)=AX*X(K) + BX           |
    +      *   .NOT.LOG         *+++++    |                                 |
    +      *XSW                *          |                                 |
    +         *  .   *   .  *              ---------------------------------
    +             F  |                                   +
    +                |+++++++++++++++++++++++++++++++++++++
    +           .  *  .  *  .              ---------------------------------
    +       . *      IF       * .   T     |  YPOS(1)=AY*ALOG10(Y(K))        |
    +      *   LOGYSW           *+++++    |  + BY                           |
    +         *  .   *   .  *              ---------------------------------
    +             F  |                                   +
    +                |+++++++++++++++++++++++++++++++++++++
    +           .  *  .  *  .              ---------------------------------
    +       . *      IF       * .   T     |  YPOS(1)=AY*Y(K) + BY           |
    +      *   .NOT.LOG         *+++++    |                                 |
    +      *YSW                *          |                                 |
    +         *  .   *   .  *              ---------------------------------
    +             F  |                                   +
    +                |+++++++++++++++++++++++++++++++++++++
    +            ---------
    +            |75     |
    +            ---------
    +
```

```
    +
 2 +       CONTINUE
    +          |
    +       . * . * .          ---------------------------
    +    * . *   IF   * . T   |   GO TO 3                |
    +      OFF        *+++++  |                          |
    +      * .       . *       ---------------------------
    +         *  |  *
    +         F  |
    +
    +      CALL PTEXT(IGDSC,ISYM,1,NULLV,NULLV,1,XPOS(1),YPOS(1))
 3 ++++++++CONTINUE
               |
            ------
            |100 |
            ------

 5         CONTINUE
           ASSIGN 80 TO M
               |
           ----------------------------------------
           | J1    =1                             |
           | N     =1                             |
           | NL    =NP                            |
           ----------------------------------------
               |
          . * . * . * .
       *  .   IF       .  *
       *   (NP-IPT)       *
     |  * .              . *  |
     |     * .  *  .  * .     |
     -          0 |           +
   ------     ------       ------
   |10  |     |10  |       |20  |
   ------     ------       ------

               |
10         ----------------------------------------
           | J2    =NP                            |
           ----------------------------------------
               |
            ------
            |70  |
            ------

               |
20         ----------------------------------------
           | N     =2                             |
           | J2    =IPT                           |
           ----------------------------------------
               |
            ------
            |70  |
            ------

               |
40         ----------------------------------------
           | NL    =NL-IPT                        |
           | J1    =J2+1                          |
           ----------------------------------------
               |
          . * . * . * .
       *  .   IF       .  *
       *   (NL-IPT)       *
     |  * .              . *  |
     |     * .  *  .  * .     |
     -          0 |           +
   ------     ------       ------
   |60  |     |50  |       |65  |
   ------     ------       ------
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
50    | N      =1                                  |

         |65   |

60    | J2     =NP                                 |

      | N      =1                                  |

         |70   |

65    | J2     =J2+IPT                             |

70    | K      =J1-1                               |
      | L      =J2-J1+1                            |

          + DO LOOP TO  +
 +++++++++  STMT #    80 +
 +        + I=1,L       +

      | K      =K+1                                |

           IF          T    | XPOS(I)=AX*ALOG10(X(K))  |
         LOGXSW        *++++| + BX                     |
            F

           IF          T    | XPOS(I)=AX*X(K) + BX     |
         .NOT.LOG      *++++|                          |
        *XSW
            F

           IF          T    | YPOS(I)=AY*ALOG10(Y(K))  |
         LOGYSW        *++++| + BY                     |
            F

           IF          T    | YPOS(I)=AY*Y(K) + BY     |
         .NOT.LOG      *++++|                          |
        *YSW
            F
```

```
75        CCNTINUE
               IF X(K) .LT. XMIN .OR. X(K) .GT. XMAX    T    GO TO 78
               F
               IF Y(K) .GE. YMIN .AND. Y(K) .LE. YMAX   T    GO TO 72
               F
78        CONTINUE
          CFF  =.TRUE.
               IF OFSC     T    GO TO 79
               F
          OFSC =.TRUE.
          CALL PTEXT(IGDSC,ERR3,48,NULLV,NULLV,1,0.0,YLL-7.0*VSPB)
               IF ERRHD    T    CALL PTEXT(IGDSC,PLTID,1
                                4,NULLV,NULLV,1,0.,YLL-5.5
               F
          ERRHD=.FALSE.
79        WRITE(NOUT,1000)
          X(K),Y(K)
72        GC TO M,(2,80)
80        CCNTINUE
90        CALL PPNT(IGDSC,XPOS,YPOS,NULLV,NULLV,1,L,1,1)
          GC TO(100,40),N
100       CCNTINUE
      C   CREATE ZERO POSITICN LINES IF ON GRAPH
      C   X(O)=BX AND Y(O)=BY FOR LINEAR GRAPHS
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
              IF                        GO TO 101
           LOGXSW      T
              F

              IF                        GO TO 101
           BX .LE.     T
  XLL  .OR. BX .GE. XU
  R
              F

          CALL STPOS(IGDSC,BX,YLL)
          CALL PLINE(IGDSC,BX,YUR)
  101
              IF                        GO TO 102
           LOGYSW      T
              F

              IF                        GO TO 102
           BY .LE.     T
  YLL  .OR. BY .GE. YU
  R
              F

          CALL STPOS(IGDSC,XLL,BY)
          CALL PLINE(IGDSC,XUR,BY)
  102     CONTINUE

          | NP    =ISAVE                          |

          RETURN
                FORMAT 1000 FORMAT(' ***OFF-SCALE POINT    X = ',E12.5,5X,'Y = ',F12.5)
                FORMAT 2000 FORMAT(' ***MAX NUMBER POINTS - ',I4,' - EXCEEDED')

      END
```

```
SUBROUTINE PRECHK(DRGPM,DRHP,DRSF,VWIN,SGRT,NATRH)

          C  CHECKS PRECIPITATION-STREAMFLOW ANOMALIES AND ADJUSTS PRECIPITATION

          C    WHERE NECESSARY

               DIMENSION  DRGPM(366),DRHP(366,24),DRSF(366)
               INTEGER   DAY,HOUR,SGRT

               ------------------------------------------
               | AHP   =0.0                             |
               | NRHA  =24-NATRH                        |
               | RGPM  =DRGPM(90)                       |
               | DAY   =90                              |
               | RMWR  =1.25                            |
               ------------------------------------------
                             |
               ------------------------------------------
 100           | DAY   =DAY+1                           |
               ------------------------------------------
                             |
                             .                 ------------------
                   .  *  *       *  *          |                |
               * .  *       IF       *  .   T  |  RMWR = 2.00   |
               *       DAY  .GT.            *++++                |
               *  200  .OR.  VWIN  .GT.*       |                |
               *  750.0                 *      |                |
                   *  .          .  *          ------------------
                         *   .   *
                         F   |+++++++++++++++++++++++++++++++++++++++
                             |
               ------------------------------------------
               | RFRISE =(DRSF(DAY)-DRSF(DAY-1))        |
               |        /VWIN                           |
               ------------------------------------------
                             |
                    +-------------------+
                    +   DO LOOP  TO     +
          ++++++++++++  STMT #    101   +
          +         +   HOUR  =  1,24   +
          +         +-------------------+
          +                  |
          +                  .                 ------------------
          +         .  *  *       *  *         |                |
          +       *  .  *   IF       *  .  T   | RGPM = DRGPM(DAY)
          +     *       HOUR  .EQ          *++++                |
          +     *.  SGRT+1                 *   |                |
          +         *  .          .  *         ------------------
          +               *   .   *
          +               F   |++++++++++++++++++++++++++++++++++
          +                   |
          +    ------------------------------------------
          +    | AHP   =AHP+DRHP(DAY,HOUR)*RGPM         |
          +    ------------------------------------------
          +                   |
          +                   .                ------------------
          +         .  *  *       *  *         |                |
          +       *  .  *   IF       *  .  T   |  GO  TO  101   |
          +     *       HOUR  .NE          *++++                |
          +     *.  NRHA                   *   |                |
          +         *  .          .  *         ------------------
          +               *   .   *
          +               F   |
          +    ------------------------------------------
          +    | RWRAIN =AHP                            |
          +    ------------------------------------------
          +                   |
          +    ------------------------------------------
          +    | AHP   =0.0                             |
          +    ------------------------------------------
          +                   |
 101      ++++++++CONTINUE
                              |
```

```
                 .
             . * . * .
         . *    IF      * .   T
       *    RFRISE .       *++++ | GO TO 102
       *GT. RWRAIN .AND. RF*
       *RISE .GT. 0.1      *
         * .            . *
             * . *
              F |

(RWRAINIF((RWRAIN .GT. RMWR .AND. RFRISE .LT. 0.02*RWRAIN) .OR. (RWRAIN
        .GT. 3.00 .AND. RFRISE .LT. 0.05*RWRAIN)) GO TO 104
                |
              |108 |


                |
102          . * . * .
         . *    IF      * .   T
       *    RWRAIN .       *++++ | GO TO 103
       *GT. 0.05           *
         * .            . *
             * . *
              F |

       | RAA   =RFRISE*2.0-RWRAIN+1.0
       | DRHP(DAY,13) =RAA
                |
       WRITE(6,1)
       DAY, RAA
                |                               FORMAT     1 FORMAT(/10X,'FOR DAY',I4,
                |                                          1X,'RAIN ADDED OF',F7.2)
              |108 |


                |
103    | RAM   =2.0*RFRISE/RWRAIN                 |
                |
              |105 |

                |
104    | RAM   =10.0*RFRISE/RWRAIN                |
                |
105          . * . * .
         . *    IF      * .   T
       *    RAM  .LT.      *++++ | GO TO 108
       * 0.0               *
         * .            . *
             * . *
              F |

       WRITE(6,2)
       DAY,RAM,RWRAIN
                |
```

```
FORMAT     2 FORMAT(/5X,'FOR DAY',I4,1X,'RAIN ADJUSTMENT MULTIPLIER IS',F8.4,
            1X,'RECORDED RAIN IS',F7.2)
                          |
                 +-------------+
                 + DO LOOP TO  +
      +++++++++++ STMT #   106 +
      +          + HOUR = 1,NRHA+
      +          +-------------+
      +                 |
      +                 |
106   ++++++++DRHP(DAY,HOUR) = DRHP(DAY,HOUR)*RAM
                        |
                  . * .  * .
              * .      IF     * .  T    ----------------------------
            *     NATRH  .E        *++++  |     GO TO 108            |
            *Q. 0                  *      |                          |
              * .              * .         ----------------------------
                  . *  . *
                   F |
                     |
           -------------------------------------
           | NFRHA=NRHA+1                      |
           -------------------------------------
                     |
                 +-------------+
                 + DO LOOP TO  +
      +++++++++++ STMT #   107 +
      +          + HOUR = NFRHA,24
      +          +-------------+
      +                 |
      +                 |
107   ++++++++DRHP(DAY-1,HOUR) = DRHP(DAY-1,HOUR)*RAM
                        |
108                     |
                  . * .  * .
              * .      IF     * .  T    ----------------------------
            *     DAY  .NE.        *++++  |     GO TO 100            |
            * 273                  *      |                          |
              * .              * .         ----------------------------
                  . *  . *
                   F |

           RETURN

END
```

```
SUBROUTINE PREPRD(RGPM,DRHP,DAY,HOUR,DPY,PRD,PEP,PRH)

      C  DIVIDES HOURLY PRECIPITATION TOTALS AMONG PERIODS FOR SMALL BASINS

         DIMENSION DRHP(366,24), PE4P(4)
         INTEGER DAY,DPY,HOUR,PRD

         | PEP  =0.0                                |

              IF                 T
           PRH .EQ.          *++++|   RETURN        |
         * C.0
              F

              IF                 T
           PRD .EQ.          *++++|   GO TO 100     |
         * 1
              F

         | PEP  =PE4P(PRD)                          |

         RETURN

100      | LHOUR=HOUR-1
         | LDAY =DAY                                |

              IF                 T
          LHOUR .G           *++++|   GO TO 101     |
         *E. 1
              F

         | LHOUR=24
         | LDAY =DAY-1                              |

              IF                 T
           LDAY .EQ          *++++|   LDAY = 365    |
         *. 0
              F

              IF                 T
           LDAY .EQ          *++++|   LDAY = 59     |
         *. 365
              F
```

```
                         |
                         .
                   .  *  .  *  .            ---------------------------------------------
             .  *        IF        *  .  T  |            LDAY = 366                      |
          *     LDAY  .EQ               *++++|                                            |
          *.  59  .AND.  DPY   .EQ.*         |                                            |
          *  366                    *        |                                            |
             *    .         .    *          ---------------------------------------------
                   *   .   *
                   F  |
                      |++++++++++++++++++++++++++++++++
        ------------------------------------------------
101     | PRLH =RGPM*DRHP(LDAY,LHOUR)                  |
        | NHOUR=HOUR+1                                 |
        | NDAY =DAY                                    |
        ------------------------------------------------
                         |
                         .
                   .  *  .  *  .            ---------------------------------------------
             .  *        IF        *  .  T  |            GO TO 102                       |
          *       NHOUR  .L             *++++|                                            |
          *E.  24                       *    |                                            |
             *    .         .    *          ---------------------------------------------
                   *   .   *
                   F  |
        ------------------------------------------------
        | NHOUR=1                                      |
        ------------------------------------------------
                         |
        CALL DAYNXT(NDAY,DPY)
        ------------------------------------------------
102     | PRNH =RGPM*DRHP(NDAY,NHOUR)                  |
        ------------------------------------------------
                         |
                         .
                   .  *  .  *  .            ---------------------------------------------
             .  *        IF        *  .  T  |            GO TO 103                       |
          *         PRH  .GT.           *++++|                                            |
          *  PRLH  .AND.  PRH  .GT*          |                                            |
          *.  PRNH                   *       |                                            |
             *    .         .    *          ---------------------------------------------
                   *   .   *
                   F  |
                   -------
                   |104  |
                   -------

                         |
        ------------------------------------------------
103     | PE4P(1)  =0.10                               |
        | PE4P(2)  =0.28                               |
        | PE4P(3)  =0.46                               |
        | PE4P(4)  =0.16                               |
        ------------------------------------------------
                         |
                   -------
                   |108  |
                   -------

104                      |
                         .
                   .  *  .  *  .            ---------------------------------------------
             .  *        IF        *  .  T  |            GO TO 105                       |
          *         PRH  .LT.           *++++|                                            |
          *  PRLH  .AND.  PRH  .LT*          |                                            |
          *.  PRNH                   *       |                                            |
             *    .         .    *          ---------------------------------------------
                   *   .   *
                   F  |
                   -------
                   |106  |
                   -------
```

```
                          |
                ---------------------------------------
105             | PE4P(1) =0.28                       |
                | PE4P(2) =0.10                       |
                | PE4P(3) =0.16                       |
                | PE4P(4) =0.46                       |
                ---------------------------------------
                          |
                      ------
                      |108 |
                      ------


                          |
106                       *
                    .  *     *  .
               .  *      IF       *  .   T     ------------------------------
              *     PRNH .GE             *++++|       GO TO 107            |
              *.  PRLH                   *    |                            |
                  *  .            .  *        ------------------------------
                      *   .   *
                        F |
                          |
                ---------------------------------------
                | PE4P(1) =0.46                       |
                | PE4P(2) =0.16                       |
                | PE4P(3) =0.28                       |
                | PE4P(4) =0.10                       |
                ---------------------------------------
                          |
                      ------
                      |108 |
                      ------


                          |
                ---------------------------------------
107             | PE4P(1) =0.10                       |
                | PE4P(2) =0.28                       |
                | PE4P(3) =0.16                       |
                | PE4P(4) =0.46                       |
                ---------------------------------------
                          |
                   +-------------+
108  +++++++++++++ + DO LOOP TO  +
     +             + STMT #  109 +
     +             + KPRD = 1,4  +
     +             +-------------+
     +                    |
     +          ---------------------------------------
109  ++++++++++ | PE4P(KPRD) =PE4P(KPRD)*PRH          |
                | PEP  =PE4P(1)                       |
                ---------------------------------------
                          |
               RETURN

END
```

```
SUBROUTINE RTVARY(CTRI,SATRI,BTRI,CHCAP,NBTRI,MXTRI,NCTRI,EXQPV,
          LSHFT,TFCFS)
        DIMENSION  AWSBIT(99),BTRI(99),CTRI(99),SATRI(99)
        LOGICAL  LSHFT
                 |
           +-------------+
           +  DO LOOP TO  +
+++++++++++++ STMT #   100 +
+          + KIA = 1,MXTRI+
+          +-------------+
+                |
+                |
+      -----------------------------------------
+      | SATRI(KIA) =0.0                       |
+      -----------------------------------------
+                |
+      -----------------------------------------
100 +++++++| AWSBIT(KIA) =0.0                  |
       | LSHFT=.FALSE.                         |
       | FMXTRI  =MXTRI                        |
       | FNBTRI  =NBTRI                        |
       | FNPTRI  =NCTRI                        |
       | TFX   =TFCFS                          |
       | TFMRT=0.1*CHCAP                       |
       -----------------------------------------
                 |
                 .                 -------------------------
            . * * * .              |                       |
         . *    IF     * .   T     |     TFX = TFMRT       |
       *    TFX .LT.         *++++ |                       |
       * TFMRT                 *   |                       |
         * .           . *         -------------------------
            * . *
              F |                             +
                |++++++++++++++++++++++++++++++
                 .                 -------------------------
            . * * * .              |                       |
         . *    IF     * .   T     |     RETURN            |
       *    FNPTRI .         *++++ |                       |
       *EQ. FMXTRI  .AND. TF*      |                       |
       *X .EQ. TFMRT           *   |                       |
         * .           . *         -------------------------
            * . *
              F |                             +
                |++++++++++++++++++++++++++++++
       -----------------------------------------
       | FNTRI=FNBTRI*(CHCAP/TFX)**EXQPV       |
       |      +0.5                             |
       -----------------------------------------
                 |
                 .                 -------------------------
            . * * * .              |                       |
         . *    IF     * .   T     |     FNTRI = 1.01      |
       *    FNTRI  .L        *++++ |                       |
       *T. 1.0                 *   |                       |
         * .           . *         -------------------------
            * . *
              F |                             +
                |++++++++++++++++++++++++++++++
       -----------------------------------------
       | NCTRI=FNTRI                           |
       | FNSTRI =NCTRI                         |
       -----------------------------------------
                 |
                 .                 -------------------------
            . * * * .              |                       |
         . *    IF     * .   T     |     LSHFT = .TRUE.    |
       *    FNSTRI .         *++++ |                       |
       *NE. FNPTRI             *   |                       |
         * .           . *         -------------------------
            * . *
              F |                             +
                |++++++++++++++++++++++++++++++
```

C-5

```
                    .
              . * * IF * .  T    +----------------------+
        *  .NOT. LS      *+++++  |   RETURN             |
        *HFT                 *   |                      |
              * .  * . *  *      +----------------------+
                 F |
                   |+++++++++++++++++++++++++++++++++++++++++
                    .
              . * * IF * .  T    +----------------------+
        *     FNPTRI .       *++++|   GO TO 101          |
        *GT. 98.5            *   |                      |
              * .  * . *  *      +----------------------+
                 F |
                   |
        +--------------------------------------------+
        | FCNTRI =ABS(FNSTRI-FNPTRI)                 |
        +--------------------------------------------+
                   |
              . * * IF * .  T    +----------------------+
        *     FCNTRI .       *++++|   GO TO 101          |
        *LE. 1.1             *   |                      |
              * .  * . *  *      +----------------------+
                 F |
                   |
              . * * IF * .  T    +----------------------+
        *     FNSTRI .       *++++|   FNSTRI = FNPTRI + 1.0 |
        *GT. FNPTRI          *   |                      |
              * .  * . *  *      +----------------------+
                 F |
                   |+++++++++++++++++++++++++++++++++++++++++
              . * * IF * .  T    +----------------------+
        *     FNSTRI .       *++++|   FNSTRI = FNPTRI - 1.0 |
        *LT. FNPTRI          *   |                      |
              * .  * . *  *      +----------------------+
                 F |
                   |+++++++++++++++++++++++++++++++++++++++++
        +--------------------------------------------+
        | NCTRI=FNSTRI                               |
        +--------------------------------------------+
                   |
101     +--------------------------------------------+
        | KB1  =0                                    |
        | KB2  =1                                    |
        | KB3  =0                                    |
        +--------------------------------------------+
                   |
102     +--------------------------------------------+
        | KB1  =KB1+1                                |
        +--------------------------------------------+
                   |
              . * * IF * .  T    +----------------------+
        *     KB1 .GT.       *++++|   GO TO 105          |
        * ABTRI              *   |                      |
              * .  * . *  *      +----------------------+
                 F |
                   |
        +--------------------------------------------+
        | KB4  =0                                    |
        | WSBIT=BTRI(KB1)/FNSTRI                     |
        +--------------------------------------------+
                   |
103     +--------------------------------------------+
        | KB4  =KB4+1                                |
        +--------------------------------------------+
                   |
```

```
                   |
                   .
               . *   * .
           . *    IF     * .    T
         *    KB4 .GT.          *++++|  GO TO 102       |
         * NCTRI                *
           * .               . *
               * .       . *
                 F   |
                     |
        | AWSBIT(KB2) =AWSBIT(KB2)+WSBIT        |
        | KB3  =KB3+1                           |
                     |
                     .
               . *   * .
           . *    IF     * .    T
         *    KB3 .LT.          *++++|  GO TO 104       |
         * NBTRI                *
           * .               . *
               * .       . *
                 F   |
                     |
        | KB3  =0                               |
        | KB2  =KB2+1                           |
                     |
 104             |103  |

 105                 |
                     .
               . *   * .
           . *    IF     * .    T
         *    ENPTRI .          *++++|  GO TO 108       |
         *GT. 98.5              *
           * .               . *
               * .       . *
                 F   |
                 +-------------+
                 + DO LOOP TO  +
      +++++++++++ STMT #   107 +
      +          + KB6 = 1,NCTRI+
      +          +-------------+
      +                 |
      +          +-------------+
      +          + DO LOOP TO  +
      +++++++++++ STMT #   106 +
      +          + KB7 = 1,KB6 +
      +          +-------------+
      +                 |
 106  ++++++++| SATRI(KB6) =SATRI(KB6)+AWSBIT(KB7)  |
      +       |       -CTRI(KB7)                    |
      +                 |
 107  ++++++++CONTINUE
                        |
 108             +-------------+
                 + DO LOOP TO  +
      +++++++++++ STMT #   109 +
      +          + KB5 = 1,MXTRI+
      +          +-------------+
      +                 |
 109  ++++++++| CTRI(KB5) =AWSBIT(KB5)              |
                        |
            RETURN

  END
```

```
      SUBROUTINE SGRID
      COMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
     1     NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
     2     IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
     3     VSPB   ,VSPL   ,NCUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
     4     XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
     5     YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
     6     AY     ,BY     ,YLPOS  ,
     7     LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
     8     FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      LOGICAL*1 GSPIN, DUM(2)
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      REAL*8 WORK(36)
      LOGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      LOGICAL*1 FA,FB,FC,GMODE1
      EQUIVALENCE (RWORK1,DEL)
C     CREATE GRID BOX ELEMENTS
      CALL STPOS(IGDSC,XLL,YLL)
      CALL PLINE(IGDSC,XUR,YLL)
      CALL PLINE(IGDSC,XUR,YUR)
      CALL PLINE(IGDSC,XLL,YUR)
      CALL PLINE(IGDSC,XLL,YLL)
      DEL  =.5*VSPB
C     CREATE LOWER TIC MARK ELEMENTS
      CALL PSGMT(IGDSC,XTIC,YLL-DEL,XTIC,YLL+DEL,NULLV,NULLV,1,NXD,
     1     1,0,1,0)
C     CREATE UPPER TIC MARK ELEMENTS
      CALL PSGMT(IGDSC,XTIC,YUR-DEL,XTIC,YUR+DEL,NULLV,NULLV,1,NXD,
     1     1,0,1,0)
C     CREATE LEFT TIC MARK ELEMENTS
      CALL PSGMT(IGDSC,XLL-DEL,YTIC,XLL+DEL,YTIC,NULLV,NULLV,1,NYD,
     1     0,1,0,1)
C     CREATE RIGHT TIC MARK ELEMENTS
      CALL PSGMT(IGDSC,XUR-DEL,YTIC,XUR+DEL,YTIC,NULLV,NULLV,1,NYD,
     1     0,1,0,1)
      RETURN
      END
```

```
SUBROUTINE SHIFT1

      CCMMON/GSPD/WORK   ,GSP1   ,UNITN  ,I2250  ,IATN   ,OAREA  ,
           NULLV  ,IGDSC  ,IPTNOW,MAXPT  ,IPT    ,ISYM   ,RWORK1,
           IGDS4  ,IGDS5  ,IGDS6  ,IDSP   ,RU     ,HSPB   ,HSPL   ,
           VSPB   ,VSPL   ,NOUT   ,NXD    ,NYD    ,XSIZE  ,YSIZE  ,
           XLL    ,YLL    ,XUR    ,YUR    ,XTIC   ,YTIC   ,XDVAL  ,
           YDVAL  ,IKEY1  ,IKEY2  ,IKEY3  ,IKEY4  ,AX     ,BX     ,
           AY     ,BY     ,YLPOS  ,
           LOGXSW,LOGYSW,NPTF   ,NRGF   ,ERRHD  ,FA     ,FB     ,
           FC     ,GMODE1,GSPIN  ,DUM    ,IRECAL
      LCGICAL*1 GSPIN, DUM(2)
      REAL*4 XTIC(6),YTIC(6),XDVAL(6),YDVAL(6)
      INTEGER*4 UNITN,OAREA,NULLV(1),GSP1
      LOGICAL*1 FA,FB,FC,GMODE1
      LCGICAL*1 LOGXSW,LOGYSW,NPTF,NRGF,ERRHD
      REAL*8 WORK(36)

      | IABSDP =IABS(IDSP)+1                     |

            IF                 T   | CALL OMIT(IGDSC)              |
         IRECAL .         *+++++
      *EQ. 0
            F

      CALL EXEC(IGDSC)
      CALL BUFOUT(I2250,IABSDP,IRC)

            IF                 T   | WRITE(NOUT,1040) IRC          |
         IRC .NE.         *+++++
      * C
            F

            IF                 T   | GO TO 17                      |
         IRECAL .         *+++++
      *EQ. 0
            F

            IF
         (IDSP)
       -        0        +
     |11   |  |12   |  |11   |

 11   CALL INCL(IGDS5,NULLV,IKEY3)
 12   CALL ATTEN
      CALL OMIT(IGDS5,NULLV,IKEY3)
 17   CALL RESET(IGDSC)

      | IDSP =IABSDP                            |

 14   RETURN
                                FORMAT  1040 FORMAT(' BUFFER READ
                                             ERROR - R.C. = ',Z8)
END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
SUBROUTINE SNCMEL(BCDFSM,SPTWCC,SPM,ELDIF,DAY,SPBFLW,XDNFS,FFOR,
            FFSI,MRNSM,DSMGH,SDEPTH,STMD,PXCSA,HOUR,SAX,SOFRF,OFRFIS,SOFRFI,
            AMFSIL,PRH,SPTW,TANSM,SPLW,SFMD,OFRF,WT4AM,WT4PM,ASM,ASMRG,
            SASFX,SARAX,DMXT,DMNT,RICY,FIRR)
      C  SNOWMELT COMPUTATION

         DIMENSION  DMNT(366),DMXT(366),FIRR(15),RICY(37)
         INTEGER  DAY,HOUR
         REAL  MHSM,MRNSM
                   |
                   .                   -------------------------------
            . * * * .                  |                             |
        . *     IF     * .   T         |     GO TO 100               |
      *   (DAY .NE          *++++      |                             |
      *. 274) .OR. (HOUR .*            |                             |
      *NE. 1)                *         |                             |
        * .               . *          -------------------------------
            * . *
              F |
                |
      ----------------------------
      | SPLW =0.0                |
      | XELR =0.0                |
      | SDSC =0.0278             |
      | FDSC =0.0                |
      | FTA  =0.0                |
      | RICD =0.0                |
      | KRIA =0                  |
      ----------------------------
                |
100      CONTINUE
      C  CALCULATION OF HOURLY AIR TEMPERATURE

      C    DMXT CURRENT DAY, DMNT NEXT DAY
                   |
                   .                   -------------------------------
            . * * * .                  |                             |
        . *     IF     * .   T         |     GO TO 101               |
      *    HOUR .NE         *++++      |                             |
      *. 4                  *          |                             |
        * .               . *          -------------------------------
            * . *
              F |
                |
      ----------------------------
      | FDSC =0.0                |
      | FTA  =FDSC               |
      | WT4PM=DMXT(DAY)-4.0*ELDIF+(XELR
      |       /4.0)*0.7*ELDIF    |
      ----------------------------
                |
101                .                   -------------------------------
            . * * * .                  |                             |
        . *     IF     * .   T         |     SDSC = -0.0278          |
      *    HOUR .EQ         *++++      |                             |
      *. 10                 *          |                             |
        * .               . *          -------------------------------
            * . *                                     +
              F |+++++++++++++++++++++++++++++++++++++++
                .                      -------------------------------
            . * * * .                  |                             |
        . *     IF     * .   T         |     SDSC = 0.0278           |
      *    HOUR .EQ         *++++      |                             |
      *. 22                 *          |                             |
        * .               . *          -------------------------------
            * . *                                     +
              F |+++++++++++++++++++++++++++++++++++++++
                .                      -------------------------------
            . * * * .                  |                             |
        . *     IF     * .   T         |     GO TO 102               |
      *    HOUR .NE         *++++      |                             |
      *. 16                 *          |                             |
        * .               . *          -------------------------------
            * . *
              F |
```

```
              |
     ----------------------------------------
     | NDAY =DAY+1                          |
     ----------------------------------------
              |
            . * .                          ----------------------------
         . *   IF  * .     T               |      NDAY = 1            |
       *    NDAY .EQ     *++++             |                          |
       *. 366            *                 |                          |
         * .           . *                 ----------------------------
            * . *
          F   |++++++++++++++++++++++++++++++++
              |
            . * .                          ----------------------------
         . *   IF  * .     T               |      NDAY = 366          |
       *    NDAY .EQ     *++++             |                          |
       *. 60 .AND. DMXT(366*               |                          |
       *) .NE. 0.0       *                 |                          |
         * .           . *                 ----------------------------
            * . *
          F   |++++++++++++++++++++++++++++++++
              |
            . * .                          ----------------------------
         . *   IF  * .     T               |      NDAY = 60           |
       *    NDAY .EQ     *++++             |                          |
       *. 367            *                 |                          |
         * .           . *                 ----------------------------
            * . *
          F   |++++++++++++++++++++++++++++++++
              |
     ----------------------------------------
     | WT4AM=DMNT(NDAY)-(XELR/4.0)*3.3      |
     |       *ELDIF                         |
     ----------------------------------------
              |
102         . * .                          ----------------------------
         . *   IF  * .     T               |      GO TO 103           |
       *    PRH .LE.     *++++             |                          |
       * 0.0 .OR. XELR .GE.*               |                          |
       * 4.0             *                 |                          |
         * .           . *                 ----------------------------
            * . *
          F   |
              |
     ----------------------------------------
     | WT4AM=WT4AM-0.825*ELDIF              |
     ----------------------------------------
              |
     ----------------------------------------
     | WT4PM=WT4PM+0.175*ELDIF              |
     | XELR =XELR+1.0                       |
     ----------------------------------------
              |
103         . * .                          ----------------------------
         . *   IF  * .     T               |      GO TO 104           |
       *    PRH .NE.     *++++             |                          |
       * 0.0 .OR. XELR .LE.*               |                          |
       * 0.0             *                 |                          |
         * .           . *                 ----------------------------
            * . *
          F   |
              |
     ----------------------------------------
     | WT4AM=WT4AM+0.825*ELDIF              |
     | WT4PM=WT4PM-0.175*ELDIF              |
     | XELR =XELR-1.0                       |
     ----------------------------------------
              |
104  ----------------------------------------
     | TEH  =WT4AM+FTA*(WT4PM-WT4AM)        |
     | FDSC =FDSC+SDSC                      |
     | FTA  =FTA+FDSC                       |
     ----------------------------------------
              |
```

```
        IF
    PRH+SPTW                T      GO TO 128
  .EQ. 0.0                *++++
        F

        IF
    HOUR .NE                T      GO TO 105
  . 24                    *++++
        F

C   CALCULATION OF TIME AGING OF THE SNOWPACK

  | SAX   =SAX+1.0                           |

        IF
    SAX .GT.                T      SAX = 15.0
  15.0                    *++++
        F
        |+++++++++++++++++++++++++++++++

105
        IF
    TEH .GT.                T      GO TO 110
  32.0                    *++++
        F

C   PRECIPITATION IN FORM OF SNOW - CALCULATE INTERCEPTION DENSITY OF NEW

C     SNOW COMPACTION, AND SETTLING SNOW PACK AND THE EFFECT ON ALBEDO

        IF
    PRH .LE.                T      GO TO 110
  0.0                     *++++
        F

  | PRH   =SPM*PRH                           |
  | HSF   =PRH                               |
  | ASM   =ASM+HSF                           |
  | PRH   =(1.0-(FFSI*FFOR))*PRH             |
  | HSFRG=PRH                                |
  | ASMRG=ASMRG+HSFRG                        |
  | FSIL  =FFSI*FFOR*HSF                     |
  | AMFSIL =AMFSIL+FSIL                      |

        IF
    TEH .LE.                T      GO TO 106
  0.0                     *++++
        F

  | DNFS  =XDNFS+((0.01*TEH)**2)             |
```

ORIGINAL PAGE IS OF POOR QUALITY

```
                 |
              -------
              |107  |
              -------

                 |
106       ----------------------------------------------
          | DNFS =XDNFS                                |
          ----------------------------------------------
                 |
107              .                  ---------------------------------------
            . * IF  * .    T        |                                     |
        *   SPTW .GT         *++++  |   *  SDEPTH = SDEPTH - ((PRH        |
        *. C.O .AND. SDEPTH  *      |                                     |
        *.GT. SPTW           *      |                                     |
            * .      .  *           ---------------------------------------
               F |++++++++++++++++++++++++++++++++
          SDEPTH/SPTW)*((0.10*SDEPTH)**0.25))
                 |
          ----------------------------------------------
          | SPTW =SPTW+PRH                             |
          | SDEPTH =SDEPTH+(PRH/DNFS)                  |
          | SASFX=SASFX+PRH                            |
          ----------------------------------------------
                 |
                 .                  ---------------------------------------
            . * IF  * .    T        |                                     |
        *   SASFX .G         *++++  |   GO TO 108                         |
        *E. PXCSA            *      |                                     |
            * .      .  *           ---------------------------------------
               F |
              -------
              |109  |
              -------

                 |
108       ----------------------------------------------
          | SAX  =SAX-1.0                              |
          ----------------------------------------------
                 |
                 .                  ---------------------------------------
            . * IF  * .    T        |   SAX = 0.0                         |
        *   SAX .LT.         *++++  |                                     |
        * 0.0                *      |                                     |
            * .      .  *           ---------------------------------------
               F |++++++++++++++++++++++++++++++++
          ----------------------------------------------
          | SASFX=SASFX-PXCSA                          |
          ----------------------------------------------
                 |
109       ----------------------------------------------
          | PRH  =0.0                                  |
          ----------------------------------------------
                 |
110       CONTINUE
                 |
                 .                  ---------------------------------------
            . * IF  * .    T        |   GO TO 127                         |
        *   SPTW .LE         *++++  |                                     |
        *. 0.0               *      |                                     |
            * .      .  *           ---------------------------------------
               F |

       C  SEASCNAL MELT FACTOR ADJUSTMENT

                 |
          ----------------------------------------------
          | FCAY =DAY                                  |
          ----------------------------------------------
                 |
```

```
                 |
     ------------------------------------------
     | KAAO =KRIA                             |
     | KRIA =1.0+(FDAY/10.0)                  |
     ------------------------------------------
                 |
                 .                  ----------------------------
           . * *   * .              |                          |
       . *      IF      * .  T      |    RICD = RICY(KRIA)     |
     *      KAAO .NE         *++++  |                          |
     *. KRIA                 *      |                          |
       * .             . *          ----------------------------
            *    .   *                                   +
            F    |++++++++++++++++++++++++++++++++++++++++
                 |
                 .                  ----------------------------
           . * *   * .              |                          |
       . *      IF      * .  T      |    GO TO 111             |
     *      TEH .LE.             *++++  |                      |
     * 32.0                  *      |                          |
       * .             . *          ----------------------------
            *    .   *
            F    |
                 |
             --------
             |114  |
             --------

C  CALCULATION OF NEGATIVE MELT
                 |
111              .                  ----------------------------
           . * *   * .              |                          |
       . *      IF      * .  T      |    GO TO 112             |
     *      TANSM  .L            *++++  |                      |
     *E. 11.5*MRNSM          *      |                          |
       * .             . *          ----------------------------
            *    .   *
            F    |
                 |
                 .                  ----------------------------
           . * *   * .              |                          |
       . *      IF      * .  T      |    TANSM = TANSM + ((5.0*M |
     *      TANSM  .L            *++++  | RNSM)**(1.3 + 2.0*    |
     *T. 1.0                 *      |                          |
       * .             . *          ----------------------------
            *    .   *                                   +
            F    |++++++++++++++++++++++++++++++++++++++++
     TANSM))     |
             --------
             |113  |
             --------

                 |
112  ------------------------------------------
     | TANSM=TANSM+MRNSM                      |
     ------------------------------------------
                 |
113              .                  ----------------------------
           . * *   * .              |                          |
       . *      IF      * .  T      |    TANSM = 0.08*SPTW     |
     *      TANSM  .G            *++++  |                      |
     *T. 0.08*SPTW           *      |                          |
       * .             . *          ----------------------------
            *    .   *                                   +
            F    |++++++++++++++++++++++++++++++++++++++++
             --------
             |127  |
             --------

C  EFFECT OF RAIN ON ALBEDO
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
114     | SARAX=SARAX+PRH                    |

               IF
           SARAX .L                T
        T. PXCSA/2.0              +++++    GO TO 115
               F

        | SAX  =SAX+1.0                      |

               IF
           SAX .GT.                T
        15.0                      +++++    SAX = 15.0
               F
               ++++++++++++++++++++++++++++++

        | SASFX=0.0                          |
        | SARAX=SARAX-(PXCSA/2.0)            |

115            IF
           TEH .GT.                T
        32.0                      +++++  M   HSM = (TEH -32.0)*BUDFS
               F
               ++++++++++++++++++++++++++++++

               IF
           TEH .LT.                T
        32.0                      +++++    HSM = 0.0
               F
               ++++++++++++++++++++++++++++++

        | HSM  =HSM*RICD                     |
        | KAA  =1.0+SAX                      |

               IF
           SAX .LT.                T
        15.0                      +++++    HSM = HSM*(1.0 - ((1.0
                                           - FFOR)*FIRR(KAA)))
               F
               ++++++++++++++++++++++++++++++

               IF
           SAX .EQ.                T
        15.0                      +++++    HSM = HSM*(1.0 - ((1.0
                                           - FFOR)*FIRR(15)))
               F
               ++++++++++++++++++++++++++++++

               IF
           PRH .GT.                T
        C.0                       +++++    HSM = HSM + ((TEH - 32.
                                           0)*(PRH/144.0))
               F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
              |++++++++++++++++++++++++++++++
              .
          . * . * .              ------------------------
    . *      IF      * .  T     |   GO TO 116            |
 *     STMD  .GT          *++++ |                        |
 *. 0.3 .AND. SPTW  .L*         |                        |
 *T. SPTWCC               *      ------------------------
    * .              . *
         *  .  *
         F  |
          -------
          |117  |
          -------

            |
116   ---------------------------------
      | MHSM =HSM                     |
      | HSM  =(SPTW/SPTWCC)*HSM       |
      ---------------------------------
            |
          . * . * .              ------------------------
    . *      IF      * .  T     |   HSM = 0.1*MHSM       |
 *     HSM  .LT.              *++++ |                    |
 *  0.1*MHSM                  *     |                    |
    * .              . *         ------------------------
         *  .  *                                  +
         F  |+++++++++++++++++++++++++++++++++++++
117         .
          . * . * .              ------------------------
    . *      IF      * .  T     |   GO TO 118            |
 *     HSM  .LT.              *++++ |                    |
 *  SPTW                      *     |                    |
    * .              . *         ------------------------
         *  .  *
         F  |
      ---------------------------------
      | HSM   =SPTW                   |
      | SDEPTH =0.0                   |
      | SPTW =0.0                     |
      | SPLW =0.0                     |
      | RICD =0.0                     |
      | TANSM=0.0                     |
      | SAX   =15.0                   |
      | OFRF =SOFRF                   |
      | OFRFIS =SOFRFI                |
      ---------------------------------
            |
          -------
          |122  |
          -------

            |
118   ---------------------------------
      | SPTW =SPTW-HSM                |
      ---------------------------------
            |
          . * . * .              ------------------------
    . *      IF      * .  T     |   GO TO 122            |
 *     SFMD  .LE              *++++ |                    |
 *.  0.0                      *     |                    |
    * .              . *         ------------------------
         *  .  *
         F  |
            .
          . * . * .              ------------------------
    . *      IF      * .  T     |   GO TO 121            |
 *     SAX  .GE.              *++++ |                    |
 *  15.0                      *     |                    |
    * .              . *         ------------------------
         *  .  *
         F  |
```

```
        IF SAX .GE. 6.0      T   GO TO 119
        F
| SDEPTH =SDEPTH-(HSM/(0.5*SFMD)) |
|122 |

119     IF SAX .LE. 10.0     T   GO TO 120
        F
| SDEPTH =SDEPTH-(HSM/(0.9*SFMD)) |
|122 |

120   | SDEPTH =SDEPTH-(HSM/(0.7*SFMD)) |
|122 |

121   | SDEPTH =SDEPTH-(HSM/SFMD) |

122     CCNTINUE
        IF SPTW .LT 0.00001  T   SPTW = 0.0
        F

C   CALCULATION OF LIQUID-WATER-HOLDING CAPACITY

| SPLWC=SPBFLW*SPTW |

        IF SFMD .GT 0.6      T   SPLWC = SPBFLW*(3.0 - 3
                                 .33*SFMD)*SPTW
        F
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                         |
                         .                        -------------------------------
                    . *     * .                  |                               |
              . *       IF      * .    T         |   SPLWC  =  0.0               |
            *      SPLWC  .L          *++++      |                               |
            *T.  0.0                  *          |                               |
              * .               . *               -------------------------------
                    * .     . *
                     F   .                                      +
                         |++++++++++++++++++++++++++++++++++++++

        C  ACCCUNTING OF MELT WATER AND RAIN

                         |
                         .                        -------------------------------
                    . *     * .                  |                               |
              . *       IF      * .    T         |   GO TO 123                   |
            *      (SPLW +            *++++      |                               |
            *HSM  + PRH)  .GT.  (SP*             |                               |
            *LWC  + TANSM)            *          |                               |
              * .               . *               -------------------------------
                    * .     . *
                     F   |
                         |
                     ---------
                     |124   |
                     ---------


                         |
123           ----------------------------------------------
              | PRH   =HSM+PRH+SPLW-SPLWC-TANSM             |
              | SPLW  =SPLWC                                |
              | SPTW  =SPTW+TANSM                           |
              | TANSM=0.0                                   |
              ----------------------------------------------
                         |
                     ---------
                     |127   |
                     ---------

                         |
124                      .                        -------------------------------
                    . *     * .                  |                               |
              . *       IF      * .    T         |   GO TO 126                   |
            *      (HSM  + P          *++++      |                               |
            *RH)  .LE.  TANSM         *          |                               |
              * .               . *               -------------------------------
                    * .     . *
                     F   |
              ----------------------------------------------
125           | SPTW  =SPTW+TANSM                           |
              ----------------------------------------------
                         |
              ----------------------------------------------
              | SPLW  =SPLW+HSM+PRH-TANSM                   |
              | PRH   =0.0                                  |
              | TANSM=0.0                                   |
              ----------------------------------------------
                         |
                     ---------
                     |127   |
                     ---------

                         |
              ----------------------------------------------
126           | TANSM=TANSM-HSM-PRH                         |
              | SPTW  =SPTW+HSM+PRH                         |
              | PRH   =0.0                                  |
              ----------------------------------------------
                         |
127           CONTINUE
                         |
              ----------------------------------------------
              | HSM   =0.0                                  |
              ----------------------------------------------
                         |
```

```
C   CALCULATION OF DENSITY AND ADJUSTMENT OF OVERLAND FLOW TIME
                    |
                    .
               . *     * .
          . *      IF       * .   T    ------------------------------
        *    SDEPTH  .          *++++|    GO TO 128                  |
        *LE. 0.0  .OR.  SPTW  .*     |                               |
        *GE.  SDEPTH            *    |                               |
          * .               . *      ------------------------------
               *    .   *
                 F  |
                    |
      -------------------------------------------
      | STMD  =(SPTW+SPLW)/SDEPTH               |
      | SFMD  =SPTW/SDEPTH                      |
      | OFRF  =0.33*SOFRF                       |
      -------------------------------------------
                    |
                    .
               . *     * .
          . *      IF       * .   T    ------------------------------
        *    SPTW  .LE          *++++|    OFRF = (1.0 - (SPTW/SPT    |
        *.  SPTWCC              *    |  WCC)*0.67)*SOFRF             |
          * .               . *      ------------------------------
               *    .   *
                 F  |                          +
                    |+++++++++++++++++++++++++++
128                 .
               . *     * .
          . *      IF       * .   T    ------------------------------
        *    SDEPTH  .          *++++|    OFRF = SOFRF               |
        *LE.  0.0               *    |                               |
          * .               . *      ------------------------------
               *    .   *
                 F  |                          +
                    |+++++++++++++++++++++++++++
      -------------------------------------------
      | OFRFIS  =SOFRFI*OFRF/SOFRF              |
      -------------------------------------------
                    |
C   CALCULATICN OF GROUNDMELT
                    |
                    .
               . *     * .
          . *      IF       * .   T    ------------------------------
        *      HOUR  .N         *++++|    RETURN                     |
        *E.  12  .OR.  SPTW  .LE*    |                               |
        *.  0.0                 *    |                               |
          * .               . *      ------------------------------
               *    .   *
                 F  |                          +
                    |+++++++++++++++++++++++++++
                    .
               . *     * .
          . *      IF       * .   T    ------------------------------
        *     SPTW  .LE         *++++|    GO TO 129                  |
        *.  DSMGH               *    |                               |
          * .               . *      ------------------------------
               *    .   *
                 F  |
                    |
      -------------------------------------------
      | PRH   =PRH+DSMGH                        |
      | SPTW  =SPTW-DSMGH                       |
      -------------------------------------------
                    |
                    .
               . *     * .
          . *      IF       * .   T    ------------------------------
        *     STMD  .LT         *++++|    SDEPTH = SDEPTH -           |
        *.  0.50  .AND.  SDEPTH*     |                               |
        *  .GT.  2.0*DSMGH      *    |                               |
          * .               . *      ------------------------------
               *    .   *
                 F  |                          +
                    |+++++++++++++++++++++++++++
        2.0*DSMGH
```

```
      RETURN

129   PRH   =SPTW+PRH+SPLW
      TANSM=0.0
      RICD =0.0
      SPLW =0.0
      SDEPTH =0.0
      SPTW =0.0
      SAX  =15.0
      OFRF =SOFRF
      OFRFIS =SOFRFI

      RETURN

END
```

```
SUBROUTINE  STAT(R,S,N,AREA,MAXR,MAXS,SSQ,SSQI,SUMD,
                           SUMSQ,MEANR,MEANS,VARR,VARS,CORCF)
C
C
      INTEGER  I,N
      REAL  R(N),S(N),AREA,SUMS,SUMR,MAXS,MAXR,SSQ,SSQI,SUMD,SUMSQ
      REAL  MEANS, MEANR, VARS, VARR, CORCF
      REAL  TEMP, TEMPS, TEMPR, AS, AR, ASR
C
C
C     INPUT:  S = ARRAY OF SIMULATED READINGS
C             R = ARRAY OF OBSERVED READINGS
C             N = NUMBER OF READINGS
C             AREA = AREA OF WATERSHED
C
C     OUTPUT: SUMS = SUM OF SIMULATED READINGS
C             SUMR = SUM OF OBSERVED READINGS
C             MAXS = MAXIMUM OF SIMULATED READINGS
C             MAXR = MAXIMUM OF OBSERVED READINGS
C             SSQ = SUM SQUARED WITH WEIGHT OF 20
C             SSQI = SUM SQUARED WITH VARIABLE WEIGHT
C             SUMD = SUM OF (OBSERVED - SIMULATED) READINGS
C             SUMSQ = SQRT OF SUM OF (OBSERVED - SIMULATED)**2)
C             MEANS = MEAN OF SIMULATED READINGS
C             MEANR = MEAN OF OBSERVED READINGS
C             VARS = VARIANCE OF SIMULATED READINGS
C             VARR = VARIANCE OF ACTUAL READINGS
C             CORCF = CORRELATION COEFFICIENT
C
C
```

```
SUMS =0.
SUMR =0.
MAXS =S(1)
MAXR =R(1)
SSQ  =0.
SSQI =0.
SUMD =0.
SUMSQ=0.

C

DO LOOP TO
STMT #    100
I= 1, N

TEMPS=S(I)
TEMPR=R(I)

C
C       COMPUTE SUM OF READINGS
C

SUMS =SUMS+TEMPS
SUMR =SUMR+TEMPR

C
C       FIND MAX OF READINGS
C

MAXS =AMAX1(MAXS,TEMPS)
MAXR =AMAX1(MAXR,TEMPR)

C
C       COMPUTE SUM SQUARES
C

IF
I .EQ. 1          T    GO TO 50
F

WEIGHT =20.

IF
N .GT.            T    WEIGHT  =  WEIGHT / [illegible]
12.
F
```

```
                 ----------------------------------------------
                 | TEMP =FLOWDI(TEMPS,TEMPR,WEIGHT)           |
                 | SSQ  =SSQ+TEMP*TEMP                        |
                 | WEIGHT =AREA+10.                           |
                 ----------------------------------------------
                                 |
                                 .           ------------------------------
                  .   *    IF   *  .   T    |                            |
                * N   .GT.        *++++ |   WEIGHT  =  WEIGHT / 30   |
                * 12.                 *     |                            |
                  .   *         *  .        ------------------------------
                      F |++++++++++++++++++++++++++++++++++++++++++++++++++
                 ----------------------------------------------
                 | TEMP =FLOWDI(TEMPS,TEMPR,WEIGHT)           |
                 | SSQI =SSQI+TEMP*TEMP                       |
                 ----------------------------------------------
                                 |
          +  C
          +  C      COMPUTE SUM OF DELTAS & SUM OF DELTAS SQUARED
          +  C
                                 |
                 ----------------------------------------------
   50     +      | TEMP =ABS(TEMPR)-ABS(TEMPS)                |
                 | SUMD =SUMD+TEMP                            |
                 ----------------------------------------------
                                 |
  100  ++++++++| SUMSQ=SUMSQ+TEMP*TEMP                        |
                 ----------------------------------------------
                                 |
             C
                                 |
             C
             C      COMPUTE MEANS
             C
                                 |
                 ----------------------------------------------
                 | MEANS=SUMS/N                               |
                 | MEANR=SUMR/N                               |
                 ----------------------------------------------
                                 |
             C
             C      COMPUTE SQRT(SUM(DELTA**2)
             C
                                 |
                 ----------------------------------------------
                 | SUMSQ=SQRT(SUMSQ)                          |
                 ----------------------------------------------
                                 |
             C
             C
                                 |
                 ----------------------------------------------
                 | AS   =0.                                   |
                 | AR   =0.                                   |
                 | ASR  =0.                                   |
                 | VARS =0.                                   |
                 | VARR =0.                                   |
                 ----------------------------------------------
                                 |
```

ORIGINAL PAGE IS OF POOR QUALITY

```
      C
      C
                           I
                    +-----------+
    +++++++++++++ + DO LOOP TO  +
    +             + STMT #  200 +
    +             +  I= 1, N    +
    +               +-----------+
    +                      I
    +
    + C
    +
    + C       COMPUTE INTERMEDIATE VARIANCE & CORRELATION COEFFICIENT
    +
    + C
    +                      I
    +       ------------------------------------------
    +       | TEMPS=S(I)-MEANS                        |
    +       | TEMP =TEMPS*TEMPS                       |
    +       | AS   =AS+TEMP                           |
    +       | VARS =VARS+TEMP                         |
    +       ------------------------------------------
    +                      I
    +
    + C
    +                      I
    +       ------------------------------------------
    +       | TEMPR=R(I)-MEANR                        |
    +       | TEMP =TEMPR*TEMPR                       |
    +       | AR   =AR+TEMP                           |
    +       | ASR  =ASR+TEMPS*TEMPR                   |
    +       ------------------------------------------
    +                      I
    +       ------------------------------------------
200 ++++++++| VARR =VARR+TEMP                         |
            ------------------------------------------
                           I
      C
      C       COMPUTE VARIANCE & CORRELATION COEFFICIENT
      C
                           I
            ------------------------------------------
            | VARS =VARS/(N-1)                        |
            | VARR =VARR/(N-1)                        |
            | CORCF=ASR/SQRT(AS*AR)                   |
            ------------------------------------------
                           I
                 RETURN

    E
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FUNCTION TABLE(X,Y,N,VALX)
         DIMENSION X(1),Y(1)
      C
      C   SUBROUTINE TO PERFORM A TABLE LOOKUP AND LINEAR INTERPOLATION
      C
         YINT(X1,X2,Y1,Y2) = (Y1-Y2)*(VALX-X2)/(X1-X2) + Y2
      C
      C   TEST FOR VALX OUTSIDE RANGE OF X TABLE
                     |
                     .        ------------------------------
             . * * .   *      |     GO TO 10               |
           . *     IF     * .  T                           |
         *   VALX.GT.       *+++++                         |
         *X(1)              *  |                           |
           * .          . *   ------------------------------
               * . *
               F  |
         ------------------------------------------
         | I     =2                               |
         ------------------------------------------
                  |
               --------
               |100  |
               --------

   10    CONTINUE
                     |
                     .        ------------------------------
             . * * .   *      |     GO TO 20               |
           . *     IF     * .  T                           |
         *   VALX.LT.       *+++++                         |
         *X(N)              *  |                           |
           * .          . *   ------------------------------
               * . *
               F  |
         ------------------------------------------
         | I     =N                               |
         ------------------------------------------
                  |
               --------
               |100  |
               --------

      C
      C   VALX WITHIN RANGE OF X TABLE
   20    CONTINUE
                  |
         ------------------------------------------
         | J     =N/2                             |
         ------------------------------------------
                  |
         ------------------------------------------
         | I     =1                               |
         ------------------------------------------
                  |
```

```
                          |
                          .
                     .  *   *  .
                  .  *    IF      *  .       T    ---------------------------------
               *      VALX.GT.          *****+++++|        I=J                      |
               *X(J)                    *         |                                 |
                  *  .              .  *          ---------------------------------
                        *   .   *
                         F  |++++++++++++++++++++++++++++++++++++
    30          CONTINUE
                          |
                          .
                     .  *   *  .
                  .  *    IF      *  .       T    ---------------------------------
               *      VALX.LT.          *****+++++|        GO TO 100                |
               *X(I)                    *         |                                 |
                  *  .              .  *          ---------------------------------
                        *   .   *
                         F  |
                            |
               ------------------------------------------
               | I      =I+1                            |
               ------------------------------------------
                            |
                        --------
                        |30    |
                        --------

          C
          C      FIND INTERPOLATED VALUE
   100          CONTINUE
                            |
               ------------------------------------------
               | TABLE=YINT(X(I-1),X(I),Y(I-1),Y(I))    |
               ------------------------------------------
                            |
                RETURN
      END
```